NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

DEVELOPMENT OF AN INTERNET INTRUSION PREVENTION TOOL

by

Dagohoy Hofilena Anunciado

December 1997

Thesis Co-Advisors: Bert Lundy
Ron Broersma

Approved for public release; distribution is unlimited

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302 and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (*Leave blank*)	2. REPORT DATE December 1997	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE DEVELOPMENT OF AN INTERNET INTRUSION PREVENTION TOOL	5. FUNDING NUMBERS
6. AUTHOR Anunciado, Dagohoy Hofilena	
7. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, CA 93943-5000	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSOR/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense of the U.S. Government.

12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This thesis explores the current shortcomings in computer and Intemet security, and how the lack of user education in basic security concepts is detrimental to computer and network security. The use of cryptography and potentially expensive technical means to secure systems will fail when one neglects security education of users. The thesis addresses a portion of the security education problem by designing and developing a tool to educate users on the two major successful methods for penetrating a computer system, weak passwords and social engineering. The tool can teach users how to pick good passwords and the steps to take to prevent social engineering attacks. The tool consists of a tutorial and ends with an exam to test user comprehension conceming picking good passwords and preventing social engineering attacks.

14. SUBJECT TERMS Computer Security, Social Engineering, Intrusion Prevention	15. NUMBER OF PAGES 90
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT **UNCLASSIFIED**	18. SECURITY CLASSIFICATION OF THIS PAGE **UNCLASSIFIED**	19. SECURITY CLASSIFICATION OF ABSTRACT **UNCLASSIFIED**	20. LIMITATION OF ABSTRACT **UL**

DEVELOPMENT OF AN INTERNET INTRUSION PREVENTION TOOL

Dagohoy Hofilena Anunciado
B.S., University of California, San Diego, 1990

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SOFTWARE ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
DECEMBER 1997

Author: Dagohoy H. Anunciado
Dagohoy H. Anunciado

Approved by:
Bert Lundy, Thesis Advisor

Ron Broersma
Ron Broersma, Co-Advisor

Dan C Boger
Dan Boger, Acting Chairman, Department of Computer Science

ABSTRACT

This thesis explores the current shortcomings in computer and Internet security, and how the lack of user education in basic security concepts is detrimental to computer and network security. The use of cryptography and potentially expensive technical means to secure systems will fail when one neglects security education of users. The thesis addresses a portion of the security education problem by designing and developing a tool to educate users on the two major successful methods for penetrating a computer system, weak passwords and social engineering. The tool can teach users how to pick good passwords and the steps to take to prevent social engineering attacks. The tool consists of a tutorial and ends with an exam to test user comprehension concerning picking good passwords and preventing social engineering attacks.

TABLE OF CONTENTS

I. INTRODUCTION

Many of the world's businesses today are moving toward the use of the Internet. Many companies see the Internet playing a more fundamental role in their operations, as an inexpensive, easy-to-deploy wide area transport network (Thyfault and Davis, May 5, 1997). Fifty-one Fortune 1000 companies were asked if the Internet could replace other Wide Area Network (WAN) technologies. The response was 45% yes, 6% already happening, 27% no, and 22% maybe ("Net Perceptions", May 5, 1997). In a follow-up question, "What capabilities would the Internet need before you would use it as a WAN?" The top response of 75% of companies was the need for better security ("Net Perceptions", May 5, 1997).

To provide better security to computer systems and the Internet will require educating the users and maintainers of these services. The solution to this problem of providing better security requires that all computer users get involved. Users need to realize that attacks on computer systems and the Internet are not just attacks on individual systems, but are attacks on the entire community (Stoll, 1989). Overburdened maintainers need automated tools to help secure these services. There is a need to develop an infrastructure that has security built into it from the beginning.

This thesis will explore the current shortcomings in computer and Internet security, and how the lack of user education in basic security concepts is detrimental to computer and network security. The use of cryptography and potentially expensive technical means to secure systems will fail when one neglects security education of users.

Cryptography is a discipline of cryptology dealing with coding a message so that it can not be read, i.e., encrypt, encipher. *Cryptology* is the art and science of secret writing and consists of cryptography and cryptanalysis. *Cryptanalysis* is a discipline of cryptology that attempts to break encoded messages, without the knowledge of the encoding technique. (Lundy, November 1997)

The thesis addresses a portion of the security education problem by designing and developing a tool to educate users on the two major successful methods for penetrating a computer system or network, weak passwords and social engineering. The prevention tool will teach users how to pick good passwords and what steps to take to prevent social engineering attacks. The tool will consist of a tutorial and end with an exam to test user comprehension concerning picking good passwords and preventing social engineering attacks.

The best way to protect a system from being penetrated is by using good passwords and properly protecting passwords. This thesis provides some guidelines for choosing good passwords and for protecting passwords. The thesis also explains what attributes make up a good password and what attributes make up a weak password.

Social engineering is the process of using social interactions to obtain information about a "victim's" computer system. The advantages of social engineering for penetrating a computer system are the low cost, simple to implement, time efficient, and high success at getting around a computer system's formidable security. The thesis provides suggestions for preventing social engineering attacks.

The thesis is organized into the following chapters.

Chapter I Introduction, is the introduction of the thesis.

Chapter II Background in Computer and Network Security, presents a background computer and network security and includes a definition for computer and network security. The chapter also includes a discussion of the components that make up computer and network security, and the methods used to provide computer and network security.

Chapter III Intrusion Prevention and Detection: Current Tools and Techniques, contains a discussion of current methods used to intrude on a system's security. This chapter also discusses ways of preventing and detecting these intrusions.

Chapter IV Internet Intrusion Prevention and Detection Tool Requirements, presents the requirements for an Internet intrusion detection and prevention tool. These requirements will be a reference point for building a prevention tool.

Chapter V Design and Implementation of a Internet Prevention Tool By Educating Users, contains a description of the design and implementation of the prevention tool.

Chapter VI Conclusion and Future Research, is the final chapter and ends with a discussion of future enhancements to the current tool and tools for future research.

In addition, the thesis contains a list of references, bibliography, glossary and appendices.

Appendix A explains how to assemble the three components used to create the prevention tool. The first component is the weak password and social engineering tutorial. The second component is a multiple choice exam to test user comprehension concerning picking good passwords and preventing social engineering attacks. The final component is a driver that integrates the tutorial with the testing component.

Appendix B contains the HTML listing for the tutorial portion of the prevention tool.

Appendix C contains the HTML listing for the testing tool driver. This testing tool driver integrates the tutorial with the multiple choice testing tool. The testing tool is written using the Java programming language.

Appendix D contains the Java source code listing of the multiple choice test.

II. BACKGROUND IN COMPUTER AND NETWORK SECURITY

The basis of computer and network security is the protection of assets. These assets include the information stored in a computer system or passed through a network, and the infrastructure supporting computer systems and networks. For the purposes of this paper the definition of *computer and network security* are the tools, policies, procedures, and protocols used to protect information while being stored and processed on a computer system, and while being transmitted through a network.

The three components of computer security are secrecy, accuracy, and availability. Network security enforces the above three components with the addition of the authenticity component. Authenticity is a related variant of accuracy. (Russell and Gangemi, 1991)

Secrecy ensures that the release of information is granted to only those users who have access to the information. This access is allowed or denied by the user's security clearances. Another term used for secrecy is *confidentiality*. The use of the term secrecy applies more towards government systems that protect national defense information and highly proprietary business information. The use of the term confidentiality is more prevalent in the business setting where protecting private information and sensitive corporate data is important. Examples of private information are medical records and payroll data. Internal memos, corporate trade secrets, and corporate strategy documents are examples of sensitive corporate data. (Russell and Gangemi, 1991)

Accuracy is maintaining of information from corruption or unauthorized malicious or accidental changes. It means that the system must not corrupt the information or allow any unauthorized malicious or accidental changes to the information. Another term used for accuracy is *integrity.* (Russell and Gangemi, 1991)

Authenticity keeps users from being able to transmit forged messages over a network. It provides a way to verify the origin of the data by determining who entered or sent the data. A stronger form of authenticity is nonrepudiation. *Nonrepudiation* prevents the denial of receipt of a message by the receiver or the denial of transmission by the sender. It provides a way of recording when a message was sent or received, and who the sender and recipients are. (Russell and Gangemi, 1991), and (Stallings, 1995)

Availability is keeping information and resources available to the users. It means that the computer system's hardware and software keep working efficiently and that the system is able to recover quickly and completely if a disaster occurs. Related to availability is denial of service. *Denial of service* is the preventing or inhibiting of availability, and means that users are unable to get needed services. (Russell and Gangemi, 1991), and (Stallings, 1995)

A. COMPUTER SECURITY

For the purposes of this paper the definition of *computer security* is the tools, policies, procedures, and protocols used to protect information while being stored and processed on a computer system.

Russell and Gangemi (1991) describe four primary methods used by computer security. These four methods are as follows:

1. System Access Controls
2. Data Access Controls
3. System and Security Administrations
4. System Design

1. System Access Controls

"The first way in which a system provides computer security is by controlling access to that system." (Russell and Gangemi, 1991) This access is usually controlled by forcing one to identify oneself and having the system authenticate one's identity. In a multi-user system there are three classic methods for proving one's identity. (Russell and Gangemi, 1991)

The first method is knowing something that no one else knows. For example, having a secret password to an account that no one else knows implies ownership of the account. In the majority of cases this method provides a more than adequate means of proving one's identity. The weakness of this method is passwords may be stolen or obtained by other means. A stolen password may have come about by writing the password down and someone else reading that password. One may have given the password away. The password may be easy to guess, or found through systematic cracking methods. (Russell and Gangemi, 1991)

The second method for proving one's identity is to have something that no one else has. This may be a key, token, badge, or smart card. To have something that no one else has implies ownship of the account. The weakness with this method of proving one's identity is a key or equivalent may be stolen, lost, or duplicated. (Russell and Gangemi, 1991)

The third and final classic method used to prove identity is by biometrics. This method uses a previously stored biometric to prove one's identity. Biometrics is the use of physical or behavior traits to identify a person. These traits may include fingerprint, handprint, retina pattern, voice, signature, or keystroke pattern. Biometrics systems are quite accurate, yet on occasions, reject valid users and accept invalid ones. The problem with this system is that many people are not comfortable using it. (Russell and Gangemi, 1991)

An additional method used to prove one's identity is to used a combination of the above methods. By using a combination of methods, the weaknesses of an individual method are minimized. For example, using a combination of a password and smart card to prove one's identity would still prevent an intruder who somehow obtained the password from accessing the system, and visa versa.

The use of passwords is still the authentication tool of choice, with the use of smart cards and biometrics as secondary authentication tools (Russell and Gangemi, 1991). Most systems require one to identify themselves with a login identifier followed by a password. UNIX and Windows NT systems are example systems.

Difficult to guess passwords are the main defense against intruders. Garfinkel and Spafford (1996) explain that the best passwords are difficult to guess because they:

1. Have both uppercase and lowercase letters.
2. Have digits and punctuation characters as well as letters. Be cautious with some special characters, e.g., #, and @, for these may have special meaning to terminal emulation software (Russell and Gangemi, 1991).
3. May include some control characters and spaces. Control characters like CONTROL-S, CONTROL-H, CONTROL-/, and CONTROL-\ can cause problem with terminal emulation software (Russell and Gangemi, 1991).
4. Are easy to remember, and so need not be written down.

Easy to guess and weak passwords may have the following attributes: (UNIX Configuration Guidelines, August 1996)

1. Are in a dictionary of some language.
2. Contain a proper noun, e.g., the name of a real or fictitious character or place.
3. Are acronyms common to some specific field or profession.
4. Are a variation of a first or a last name.
5. Match the login identifier or account (Garfinkel and Spafford, 1996).
6. Contain the vender-supplied default passwords.
7. Contain no password at all.

The following are some suggestions for picking a good password:

1. Combine short words with a special character or a number, e.g., sale2noon or love-hate (Garfinkel and Spafford, 1996).
2. Choose an easy-to-remember phrase or lyric, and use the first letters to form a password. Also add punctuation or mixed case letters. For example, "Hush a my baby", would become Ham-b-. (UNIX Configuration Guidelines, August 1996).
3. Pick a nonsense word that is still pronounceable, e.g., cU2Morow or U4eyes (Russell and Gangemi, 1991).

Passwords and the file that store passwords need protection. The following are some suggestion for protecting passwords: (Russell and Gangemi, 1991), (UNIX Configuration Guidelines, August 1996), and (Garfinkel and Spafford, 1996)

1. Make sure that all logins have password.
2. Change all system, test, or guest passwords before allowing users to log in. Example system, test, or guest accounts are root, system, test, demo, and Administrator.
3. Do not ever share user passwords.
4. Do not write a password down, especially on a terminal, a computer, or anywhere around a workspace. If one does write down a password, do not identify it as a password. One may want to disguise a password by expanding it to a phrase.
5. Do not type a password while anyone is watching.
6. Do not record a password online or send a password using electronic mail. *The Cuckoo's Egg*, Clifford Stoll (1989), relates a story of how an intruder gathered valid passwords by scanning a system's email and text files for the word "password". If one does need to store passwords and accounts online or send them using email, use encryption to protect the file or message.
7. Change a password immediately if one believes the password is stolen.
8. Change a password on a regular basis, even if a password is not compromised.

2. Data Access Controls

Controlling the access to the data stored on a system is the second way that a system provides computer security. Data protection and access controls are usually not a concern in the single user environment of a PC. In a multi-user system, shared environment data protection and access controls are very important. It is common to allow other users to read one's files, but one would probably not want other users modifying or deleting files. In a multi-user environment where one trusts everyone, there is still a need to protect data from accidental modifying and deleting. (Russell and Gangemi, 1991)

Discretionary access control (DAC) and mandatory access control (MAC) are the two basic types of access controls that a system provides to protect files. Discretionary access control allows the user to decide how to protect one's files and whether to share data. Mandatory access control is more complex in that the system protects all the files. A MAC system has a label for everything. The system decides whether a user can access a file by using the security policy established by the user's organization, and comparing the label of the user with the label of the file. (Russell and Gangemi, 1991)

DAC is an access policy that restricts access to files, and other system objects such as directories and devices. The identity of the users and the groups to which they belong is the basis of this access. Application of DAC is at the user's discretion. In contrast, the system controls access in MAC. Table 2-1, show the three basic types of access that most DAC systems support.

Type	Description
Read	Access to read the file.
Write	Access to write, change, or replace the file.
Execute	Execute permission to run the program. The execute permission only makes sense if the file is a program.

Table 2-1 DAC Access Type. After Russell and Gangemi (1991).

Many systems divide the world of users into three categories: self, group, and public (Table 2-2). This method for dividing the world of users into three categories is sometimes referred to as self/group/public controls. Each of these categories assigned access determines what users can do to a file. In UNIX, there are three categories namely, user/group/other (UGO) controls. (Russell and Gangemi, 1991).

Category	Description
Self	The creator or owner of the file.
Group	A set of users. For example, all the users in a particular workgroup may be in the R&D group.
Public	Users other than those in category self and group.

Table 2-2 User Categories. After Russell and Gangemi (1991).

In UNIX, representation of each file's categories is by a set of bits called file permissions. Using the command "ls -l" one would see a list of files with their file permissions. Figure 1, is an example of file permissions for one item in a list.

```
-rw-rw-r--   1    doug    duap    4025    Sep 10 06:08    README
```

Figure 1 File Permissions. After Russell and Gangemi (1991).

The file permissions for the README file owner doug indicate that the doug can read and write the file (rw-), the duap group members also have read and write access (rw-), while everyone else only has read access (r--). The first position of the README file permission indicates the type of file. README is a plain file and indicated by the dash (-) in the first position. Other file types include, but are not limited to the following: directory (d), raw device (c), and block device (b). For more details on the different file types that the UNIX system supports, see UNIX manual for the "ls" command. The dash (-) in the rest of the file permissions is an indication of not having permission. For the README file, the owner doug and duap group members do not have permission to execute. Everyone else does not have write or read permission for README.

In general, the self/group/public controls satisfy all the needs for protecting files. There are times when one needs to allow file access in different ways for different users, or need to keep one user from accessing a file. Depending on how many special cases there are, this task is not easy to accomplish with the self/group/public controls. For this reason many vendors began offering access control lists (ACLs) in addition to their self/group/public controls. ACLs are lists of user and groups, with their specific permissions, and are a more flexible way of providing discretionary access control. One of the drawbacks in using ACLs is that each vendor implements ACLs in different ways. See the system manuals or talk with the OS vendor for more details about how the vendor implements ACLs. See also Russell and Gangemi (1991), and Garfinkel and Spafford (1996).

MAC is an access policy supported by systems that process sensitive data. Sensitive data may include government classified information or sensitive corporate data. In a MAC system all subjects and all objects must have sensitivity labels. Examples of subjects are users and programs. Files, directories, devices, windows, and sockets are examples of objects.

Sensitivity labels have two parts, a classification and a set of categories. Categories are sometimes known as compartments. The following is an example of a sensitivity label:

```
SECRET [SOME, SET, OF, CATEGORIES]
```

Classifications are arranged as levels of trust, with each level of classification having more trust than the classification beneath it. An example of classification levels, based on the Department of Defense multi-level security policy, is as follows: (Russell and Gangemi, 1991)

TOP SECRET
SECRET
CONFIDENTIAL
UNCLASSIFIED

Categories or compartments are nonhierarchical and represent separate areas of information in a system. Together, the categories make up a category or compartment set. One would use a category subset to separate information on a need to know basis. A certain category subset would prevent one with a classification higher than the classification of the information from accessing the information. For information on access decisions in a MAC system, see Russell and Gangemi, 1991.

3. System and Security Administrations

The past two sections explain how a system provides computer security. This section introduces the human side of computer security. Security is not automatic. Both the system and security administrators must work together to carry out the site's security policy. It is up to the system and security administrators to make or break the site's system security. Training users, setting up and protecting the password file and other system-critical files, and examining audit logs are some of the many ways to translate a site's security policy into human terms. (Russell and Gangemi, 1991)

Policies are the rules of conduct and behavior that arise from a consensus among a constituency. They establish the acceptable standards of behavior in one's facility and are essential for communicating consensus on an issue. A collection of written decisions on how to address particular situations are, in effect, policies. Placing this collection of decisions in an accessible location provides a way to disseminate this information, which prevents confusion and duplication of efforts among staff and users. (*A Guide to Developing Computing Policy Documents*, 1996)

For effective policy and security awareness to occur, management must lead. Security concerns and awareness by the users are important, but the users cannot build or sustain an effective culture of security. Management must treat security as important, and abide by all the same rules and regulations as everyone else. Computer security means protecting information. All plans, policies, and procedures should reflect the need to protect information in whatever form it takes. It is up to system and security administrators to work with management to set this computing and security policy. (Garfinkel and Spafford, 1996)

For information on setting up a computing and security policy see Oppenheimer, Wagner, and Crabb (1997), and Garfinkel and Spafford (1996).

4. System Design

The final way in which a system provides computer security is through the design of the system. Creating a system to meet specific security requirements enhances a system's computer security. These enhancements to computer security come about through the following: (Russell and Gangemi, 1991)

1. Support sound principles of hardware and operating system design.
2. Take advantage of the basic hardware and software security characteristics in the system design.
3. Have the ability to support specific security features in the future.

B. NETWORK SECURITY

For the purposes of this paper the definition of *network security* is the tools, policies, procedures, and protocols used to protect information while being transmitted through a network.

There are four main approaches for providing network security: (Russell and Gangemi, 1991)

1. Keep The Network Communication From Being Intercepted
2. Encrypt Data Being Sent
3. Apply Trusted System Principles
4. Configure The Network For Security

1. Keep The Network Communication From Being Intercepted

Some ways at keeping network communication from being intercepted include protecting networking equipment and choosing the most secure networking medium. Place networking equipment, such as switches and hubs, in physically secure places where intruders cannot easily attack. Choose networking cables that are difficult to tap or make it difficult to tap by placing the cables in a conduit. A fiber cable is an example of a networking medium that is difficult to tap. Placing network cables in a conduit filled with pressurized gas and a pressure sensor that detects a drop in pressure is a way of making a network medium difficult to tap. (Russell and Gangemi, 1991)

2. Encrypt Data Being Sent

Encryption is a good way to protect the secrecy of network communication. Techniques such as message authentication and digital signatures may use encryption to ensure the accuracy and authenticity of network communication. Message authentication is a procedure that can ensure accuracy and authenticity by verifying that received messages are unmodified and the messages come from the alleged source. It may also ensure accuracy by verifying that the sequencing of messages between parties is unmodified and that no delay or replay of the messages occurred. Message authentication protects two parties who exchange messages from any third party. However, it does not protect the two parties against each other. The use of a digital signature can provide the protection necessary between a sender and receiver who do not have complete trust for each other. (Russell and Gangemi, 1991), and (Stallings, 1995)

A digital signature is an authentication technique that includes measures to provide nonrepudiation between two parties. Nonrepudiation prevents the denial of receipt of a message by the receiver or the denial of transmission by the sender. A digital signature is similar to the handwritten signature and must have the following properties: (Stallings, 1995)

1. It must be able to verify the author and the date and time of the signature.
2. It must be able to authenticate the contents at the time of the signature.
3. The signature must be verifiable by third parties, to resolve disputes.

A product that provides a majority of the above techniques is Pretty Good Privacy (PGP). Phil Zimmerman created PGP and placed it on the Internet. PGP provides confidentiality and authentication services for email and file storage. Other services that PGP provides are compression, email compatibility, and segmentation. PGP provides message compression for email transmission or file storage using ZIP. ZIP is a free compression package written by Jen-lup Gailly, Mark Adler, and Richard Wales. To provide email compatibility, PGP uses radix-64 to convert encrypted messages to ASCII strings. ASCII is an acronym for American Standard Code for Information Interchange. ASCII is a standard to achieve compatibility between various types of data processing and data communications equipment (*Webster's New World Dictionary of Computer Terms*, 1983). Radix-64 conversion is a technique for mapping arbitrary binary data into printable characters. PGP provides segmentation and reassembly service to accommodate maximum email message size limitations. (Lundy, November 1997) and (Stallings, 1995)

The advantages of PGP are the application is processor independent, based on extremely good algorithms, and is not controlled by any government. Versions of PGP exist for DOS/Windows, Macintosh, and Unix. Currently, there are two versions of PGP available, a free version and commercial version. (Lundy, November 1997) and (Stallings, 1995)

Another scheme for providing secure email over the Internet is Privacy Enhanced Mail (PEM). PEM is a standard adopted by the Internet Architecture Board (IAB) to provide secure email. PEM is implemented at the application layer, allowing it to operate independent of the lower protocol layers, operating system, or host. PEM is compatible with nonenhanced mail systems and multiple mail transport facilities. It provides support for secure mailing lists and PC users. It also is compatible with various key management mechanisms. PEM provides confidentiality, authentication, message integrity, and key management services. (Schneier, 1996), and (Stallings, 1995).

For further details concerning PGP and PEM, see Stallings (1995) and Schneier (1996).

3. Apply Trusted System Principles

Applying trusted system principles means extending computer security methods to the network environment. Just as system access controls require a user to authenticate oneself before being allowed access to the system, network access controls require one to authenticate oneself before being allowed access to network services. In network access controls, a user on one system must identify and authenticate themselves before connecting to another system. The strengths that networks require users and systems to identify and authenticate themselves differ between networks. In most networks a user and system must prove their identity only once to access network services. In more highly trusted networks, a user and system must prove their identity each time they request a network service. The common way users prove their identity in a network is by using passwords. Transmitting passwords in plaintext gives an intruder the opportunity to steal these passwords. An intruder can later use these passwords to attack the network and all the hosts using the network. Network authentication systems such as Kerberos addresses this problem of plaintext authentication by using a set of encrypted keys called tickets for authentication. The Massachusetts Institute of Technology created Kerberos, which is a network authentication system based on private key cryptography (Garfinkel and Spafford, 1996). (Russell and Gangemi, 1991)

Discretionary access control (DAC) in a network environment may restrict access to the network for certain remote users and systems. Certain network services might be available only to certain groups and systems. Extending labeling and mandatory access control (MAC) to the network environment will make it possible for systems to recognize the sensitivity of data being sent among systems. MAC will allow the access of data between systems. For example, MAC keeps TOP SECRET data from being sent over a network to a system labeled as SECRET. (Russell and Gangemi, 1991)

Expanding system and security administrations to the network environment may be a large task. This task is usually large enough to create new groups of network administrators and network security officers. These groups are responsible for creating and carrying out network security policy, configuring the network and all network security features, establishing the network databases used for access control, and monitoring network traffic and performance. (Russell and Gangemi, 1991)

Just as system design enhances computer security, a network design that meets specific security requirements will also enhance network security. These enhancements to network security come about by taking advantage of the following:

1. Support sound design principles.
2. Take advantage of the basic hardware and software security characteristics in network equipment.
3. Have the ability to support specific security features in the future.

4. Configure The Network For Security

Configuring the network for security is the responsibility of the network administrator and network security office. They must work to balance the carrying out of network security policy with the design of the network.

C. COMPUTER AND NETWORK SECURITY

For the purposes of this paper the definition of *computer and network security* are the tools, policies, procedures, and protocols used to protect information being stored and processed on a computer system, and while being transmitted through a network.

Primary concern of computer security is with the protection of individual computer systems, whether or not the system is connected to a network. Primary concern of network security is with communication channels. Many of the threats to computers enter from a network channel and the endpoints of communications channels are usually computers. One can not ignore either of the concerns of computer security or network security to have a comprehensive computer and network security. (Lundy, November 1997)

The use of cryptography plays a great role in the protection of computers and networks. Governments, banks, and other organizations use cryptography to keep messages secret and protect electronic transactions from modification (Anderson, November 1994). By using cryptography to protect data and communications, individuals can protect their privacy (Schneier, 1996).

D. INTRUSION PREVENTION AND DETECTION

Webster's Ninth New Collegiate Dictionary (1991) defines "prevention" as the act of preventing or hindering.

Webster's Ninth New Collegiate Dictionary (1991) defines "detection" as the act of detecting.

Webster's Ninth New Collegiate Dictionary (1991) defines "intrusion" as the act of intruding or the state of being intruded; especially the act of wrongfully entering upon, seizing, or taking possession of the property of another.

III. INTRUSION PREVENTION AND DETECTION: CURRENT TOOLS AND TECHNIQUES

A. DEFINITION OF AN INTRUDER

An *intruder* is an entity that makes unauthorized access to a computer system or network. This entity may be a person, an automated agent, or a combination of the two.

B. METHODS USED TO INTRUDE

Effective methods used to intrude on a system's security are the use of social engineering and password cracking.

1. Social Engineering

The hacker community associates the term *Social Engineering* with the process of using social interactions to obtain information about a "victim's" computer system. This interaction may be as simple as calling a company and asking people for their passwords, possibly by impersonating a company's information security staff. Blackmail and physical threat are other methods of Social Engineering. More elaborate methods used to obtain a company's sensitive information include going through the company's garbage, posing as an employee of the company, or even becoming an employee of the company. (Winkler and Dealy, 1995)

The advantages of social engineering are low cost, simple to implement, time efficient, and high success at getting around a company's formidable security.

2. Password Cracking

Password cracking is obtaining a target host's password file and running password cracking software against the password file. This password cracking method attempts to find valid passwords for accounts in the password file. Examples of UNIX password cracking software are Crack and John the Ripper. Note that similar password cracking software is available on Windows NT. Finding the root password would be the most threatening to a system's security, since an intruder will have total access to the targeted host. One of the best defenses against password cracking is having difficult-to-guess passwords for all accounts in the password file. See section II.A.1 System Access Controls for guidelines in choosing difficult to guess passwords.

3. Stealing Superuser Privileges

Most intruding methods attempt to gain greater access to a system. These attempts may include social engineering and password cracking methods. Usually, the main objective is to gain access to a special user account that can do virtually anything to the targeted host system. In the UNIX OS, this special user account is the root account commonly referred to as the "superuser". Intruders exploit this UNIX weakness by learning the root password or gain superuser privileges by exploiting known UNIX security holes.

In Windows NT the name of the special user account is Administrator. Attack methods used on a Windows NT system would be similar to those used against a UNIX system. These attacks center on gaining access to the Administrator account or an account with the Administrator privileges.

Many methods for gaining superuser privileges are through exploiting known security vulnerabilities. A vulnerability is a point where a system is susceptible to attack. (Russell and Gangemi, 1991) Over the years people have found many vulnerabilities in

UNIX. The following areas are where some of the UNIX vulnerabilities exist or were found in the past:

1. Smashing the Stack or Buffer Overflow
2. Telnet
3. File Transfer Protocol (FTP)
4. Trivial File Transfer Protocol (TFTP)
5. Simple Mail Transfer Protocol (SMTP)
6. World Wide Web (WWW)
7. Network File System (NFS)
8. Network Information Service (NIS)
9. X Window System

a. Smashing the Stack or Buffer Overflow

Smashing the stack is a programming method that combines the C programming language with specific UNIX file system permissions to manipulate the operating system to grant intruders superuser privileges. Smith (May 7, 1997), analyzing how to construct stack smashing exploits, explains how these exploits work and what to do to prevent the exploits. Other readings on stack smashing include Instenes (April 1997), Aleph One (April 1997), and Aleph One (November 8, 1996).

b. Telnet

The telnet application lets users log in to remote hosts over TCP/IP networks. The telnet access-control mechanism relies on a password for a user account to authenticate one's identity. The transmitting of this password and account information is in cleartext. A network sniffer, installed by an intruder, is able to capture passwords by monitoring local network traffic for login activity. Note, the telnet application protocol is not inherently vulnerable, but the transporting of the password and account information in the clear is vulnerable. (Gallen and Sutterfield, November 1996)

c. File Transfer Protocol (FTP)

The FTP Application allows users to transfer files between two hosts over TCP/IP networks. FTP suffers from the same vulnerability as telnet when transmitting password and account information. Misconfigured FTP servers may allow remote access to all files on the server. Common misconfigurations include access to the system password file and granting improper user privilege to the anonymous login account. (Gallen and Sutterfield, November 1996)

Anonymous FTP Abuses (1996), provides a general overview of problems associated with abuses of anonymous FTP (File Transfer Protocol). The document provides detection, reaction, and prevention guidelines to address two issues related to anonymous FTP abuse: software piracy, and misconfigured or compromised FTP server.

d. Trivial File Transfer Protocol (TFTP)

The TFTP allows a user to transfer files from one system to another. It lacks the same degree of user authentication and file protection as FTP. General use of TFTP is in select applications, such as router software, configuration loading, or diskless workstation and X terminal boot processes. Improperly configured TFTP can provide unauthorized users access to files, which could allow an intruder to steal the password file. TFTP should be run on systems that need it for business purposes and should run in a secured mode to restrict TFTP to files in a specified directory. (Gallen and Sutterfield, November 1996)

e. Simple Mail Transfer Protocol (SMTP)

The SMTP protocol allows users to send e-mail from one host to another over TCP/IP networks. The most common implementation of an SMTP server is the UNIX sendmail program, which is a well-known source of security problems. Most vulnerabilities in SMTP servers involve tricking the server to execute the body of a mail message as a shell script. A script may be written to mail the system password to the intruder, or install backdoors for telnet access for the intruder.

f. World Wide Web (WWW)

The WWW is a system for exchanging information over the Internet. Special programs called Web servers make up the backbone of the Web. Web servers make information available on the network. Other programs called Web browsers allow one to access information stored on a Web server. (Garfinkel and Spafford, 1996)

Most Web-servers allow putting programs behind Web pages to process user-submitted data. The Common Gateway Interface (CGI) is a protocol often used to create these programs. These CGI programs may present a security risk if Web programmers use improper programming techniques.

g. Network File System (NFS)

NFS is a mechanism for sharing disk volumes between hosts over a TCP/IP network. The building block for NFS is Remote Procedure Calls (RPC's). RPC allows program distributing and makes it easy to create network-based client/server programs. Network-based client/server programs communicate with each other using remote procedure calls. An example of program distributing is taking a computationally intensive algorithm running on a high-speed computer, a remote sensing device running on another computer, and the results compile on a third. (Gallen and Sutterfield, November 1996), and (Garfinkel and Spafford, 1996)

The default means for RPC clients and servers to authenticate themselves is weak and is easy to simulate by an intruder. Each RPC request contains the UID and a set of GID's that a RPC server implicitly trusts. There are no provisions for authenticating UID and GID's on the RPC server. Little known features in the portmapper permit easy subversion of a misconfigured NFS server host. These features allow unauthorized viewing or modifying of files on exported volumes. The portmapper program provides a directory of sorts for all RPC-based services. (Gallen and Sutterfield, November 1996), and (Garfinkel and Spafford, 1996)

h. Network Information Service (NIS)

NIS lets multiple systems share the same password, group, and host files across a network. NIS was previously known as Yellow Pages. Because NIS is vulnerable to several attack methods, one should avoid using NIS. Intruders usually exploit NIS vulnerabilities to steal the password file. Cracking a password in the password file will mean access to all hosts in that NIS domain. (Gallen and Sutterfield, November 1996)

i. X Window System

The X Window System, i.e., X, is a popular network-based window system that allows may programs to share a single graphical display. X-based programs display their output in windows. This can be on the same computer that the program is running on or any other computer on the network. A special program called the X

Window Server, or X server, controls each graphical device that runs X. Other programs, called X clients, connect to the X server over the network and tell the X server what to display. (Garfinkel and Spafford, 1996)

An improperly protected X can allow X client programs to monitor information on X displays. This information may be email messages typed on a keyboard, or passwords typed on a keyboard. The two common mechanisms used to protect X is the magic cookie mechanism, and the xhost mechanism. The magic cookie mechanism uses a 128-bit "cookie". The X Display Manager (xdm) or the user creates this "cookie" at login, and stores the "cookie" in a user's .Xauthority file. Each client program reads the "cookie" from the .Xauthority file and passes it to the server when establishing the connection. This limits access to the user's display process that have access to the user's .Xauthority file. The xhost mechanism uses an access control list of all hosts allowed to access the X server. Even with the above mechanisms, a target host running X and accepting a telnet to port 6000, may be vulnerable to a denial of service attack. Some X servers read a small packet from the client before determining whether or not the client is in the xhost list. If a client connects to the X server but does not transmit this initial packet, the X server halts all operation until the X server times out in 30 seconds. Some implementations of the X server will remain frozen until the connection is aborted. (Farmer and Venema, December 2, 1993), and (Garfinkel and Spafford, 1996)

4. Security Analysis Tool for Auditing Networks (SATAN) Attacks

SATAN is a network scanning tool used to probe one's host system for well-known security vulnerabilities. SATAN scans one's host system using the network where the host resides. Scanning a host through the network is characteristic of what an outside intruder would do. Wietse Venema and Dan Farmer are the authors of SATAN. Venema and Farmer (1993) created SATAN to help system administrators determine network related security problems, reports the problems, and do so in a way that would not exploit the problems. Wietse also authored the tcpwrapper package, while Dan authored the Computer Oracle and Password System (COPS) package. Both the tcpwrapper and COPS packages protect a host from the "inside". Tcpwrappers allows one to monitor and filter incoming requests from servers started by inetd. Tcpwrappers also allows one to allow or deny access to one's host from other hosts on the network. COPS is a collection of short shell scripts and C programs that perform vulnerability checks of one's hosts. Some of the checks include bad permissions on various system files and directories, and malformed configuration files. (Garfinkel and Spafford, 1996)

The basic attack of SATAN is to use Unix utilities to gather information on a target host. Example of Unix utilities that SATAN uses are finger, showmount, and rpcinfo. The finger utility gathers information about users on a targeted host. The showmount utility finds out what files a targeted host exports. The rpcinfo utility gathers information about a targeted host's RPC services. The information gathered from these utilities cues SATAN as to what vulnerabilities may be available for exploit.

5. Trust Attacks

Farmer and Venema (1993) discusses the notion of trust in the situation when a server permits the use of a local resource by a client without password authentication when the use of password authentication is customary. The paper limits the discussion of trust to clients in disguise.

The ways the host trusts include the following:

1. .rhost and host.equiv files that allow access without password verification
2. Windows servers remote access that allow remote systems to use and abuse privileges
3. Exporting files under access control of the NFS

It is possible to spoof, fool, or subvert any form of trust. If an intruder compromises the database contained by a host, the intruder can convince the target host that he is coming from any trusted host. The compromised database may be NIS, DNS, or something else. It is possible to determine the hosts trusted by a target by the location the system administrator or system accounts last logged in from. An example of a system account on UNIX is the root account.

C. POSSIBLE WAYS OF PREVENTING AND DETECTING INTRUDING METHODS

1. Preventing Social Engineering

Winkler and Dealy (1995) suggest the following lessons learned to prevent Social Engineering:

1. Do Not Rely Upon Common Internal Identifiers
2. Implement a Call Back Procedure When Disclosing Protected Information
3. Implement a Security Awareness Program
4. Identify Direct Computer Support Analysts
5. Create A Security Alert System
6. Social Engineering to Test Security Policies

a. Do Not Rely Upon Common Internal Identifiers

Many companies rely on some type of identifier to authenticate employees, e.g., an Employee Number. Unfortunately, intruders will compile a list of valid identifiers and use these identifiers to authenticate themselves when challenged. Intruders easily obtain identifiers from real employees as part of the Social Engineering attack.

Companies should separate personnel functions from computer support functions by having a separate identifier for the two activities. A separate identifier would provide additional security to both personnel and computer activities.

b. Implement a Call Back Procedure When Disclosing Protected Information

One may have prevented many of the Social Engineering attacks by verifying the identity of the caller by calling them back at their proper telephone number, as listed in the company telephone directory. This procedure creates a minimal inconvenience to legitimate activities, while providing enhanced security for the company sensitive information. Requiring employees to call back anyone asking for personal or proprietary information minimize compromises. Use of Caller ID services might also be an acceptable solution.

c. Implement a Security Awareness Program

Computer professionals cannot assume that basic security practices are basic to non-computer professionals. A good security awareness program implemented at minimal cost can save a company from loosing millions of dollars. Without a good security awareness program, non-computer professionals would not know that giving out one's password to a stranger is a security risk to the company's information.

d. Identify Direct Computer Support Analysts

Every employee of a company should have one computer analyst be the focal point for all computer support, and should be the only person to directly contact the employee. Each analyst should have assigned no more than 60 employees, and instruct all employees to contact their analyst if ever contacted by someone claiming to be from computer support.

e. Create A Security Alert System

A security alert system should be in place to allow an employee to alert other employees of a possible intrusion. It is important to note that without this system, intruders are able to continue the same Social Engineering attacks. Security alert system is paramount to barring easy access for intruders.

f. Social Engineering to Test Security Policies

The only conceivable method to test security policies and their effectiveness is by using Social Engineering. There are many security assessments to test for physical and electronic vulnerabilities, but few assessments study the human vulnerabilities inherent in computer users. Note that only qualified and trusted people should perform these attacks.

2. Preventing Password Cracking

The best ways to prevent password cracking are to create difficult-to-guess passwords, and to protect the passwords and the file that store passwords. Refer to Section II.A.1, for suggestions on how to choose good passwords and how to protect passwords.

3. Preventing And Detecting the Stealing of Superuser Privileges

a. Preventing And Detecting Anonymous FTP Attacks

One way to prevent anonymous FTP attacks is to have a properly configured and administered FTP area. Anonymous FTP Configuration Guidelines (1995), provides guidelines to configuring an anonymous FTP area. This document breaks down the configuring and administering of FTP into three sections:

1. Configuring anonymous FTP
2. Providing writable directories in your anonymous FTP configuration
3. Related CERT Advisories

The "Configuring anonymous FTP" section suggests that the site uses the most recent version of their FTP daemon. The FTP account and associated group of the FTP account should not own the anonymous FTP directories. The document seems to recommend using the root account, the system group, and protecting the FTP root directory and FTP subdirectories so that only root account has write permission. The document strongly recommends not to use the system's /etc/passwd and /etc/group in the ~ftp/etc directory. The document recommends dummy ~ftp/etc/passwd and ~ftp/etc/group files and the passwd file should not contain account names that are the same as those in the system's /etc/passwd.

The section on "Providing writable directories in your anonymous FTP configuration," provides three suggestions for those sites that want a "drop off" directory: use a modified FTP daemon, use protected directories, and use a single disk drive. Use a modified FTP daemon that will control access to the "drop off" directory. Use protected directories only known between local users and anonymous users that wish to have "drop off" permission. This method requires prior coordination and cannot guarantee protection from unwanted use of the writable FTP area, but has been used effectively by many sites. Use a single disk drive for the writable FTP area and monitor that area on a continuing basis to ensure that it is not being misused.

The "Related CERT Advisories" section points to advisories related to FTP daemons or impact on providing FTP service.

b. Preventing SMTP Attacks

To prevent SMTP attacks, Gallen and Sutterfield (November 1996) offer the following guidelines for configuring sendmail:

1. Whenever possible, remove the prog mailer, which allows the machine to execute an e-mail message as a program
2. Disable the EXPN and VRFY commands, which may let intruders gain usernames
3. Disable the DEBUG mode and WIZ command, if applicable
4. Remove mail aliases that send to programs such as decode
5. Apply appropriate security patches
6. Upgrade to the latest version of sendmail

c. Preventing World Wide Web (WWW) Attacks

To prevent some of the security risks posed by CGI applications, Gallen and Sutterfield (November 1996) suggest avoiding the following programming techniques: calls to system() and popen(); failures to detect and filter HTML metacharacters for the newline or semicolon characters; and scripts that directly pass user-supplied input as command-line arguments. The danger is an intruder can insert their own code into a CGI script and open up security holes on the targeted system. Gallen and Sutterfield also suggest removing CGI utilities and applications that are not necessary for the continued operation of a Web site.

The Security Improvement Team Networked Systems Survivability Program has put out a booklet concerning public Web site security. This booklet is in draft form for limited external review and entitled *Security for a Public Web Site (April 21, 1997)*. *Security for a Public Web Site* recommends and describes a three-step approach for improving the security of a public Web site. The approach requires implementing security practices in three areas which Table 3-1 summarizes.

Area	**Recommend Practice**
Selecting server technology	1. Include explicit security requirements when selecting server and host technologies.
Configuring server technology	2. Isolate the Web server from the organization's internal network. 3. Maintain the authoritative copy of the web site content on a more secure host. 4. Offer only essential network services and operating system services on the server host machine. 5. Make use of available access controls for information stored on the host machine. 6. Configure the Web server software to enhance security. 7. Consider the security implications when choosing external programs that the server can execute.
Operating the server	8. Administer the Web server in a secure manner. 9. Routinely look for unexpected changes to directories and files. 10. Regularly inspect the system and network logs.

Table 3-1 Security for a Public Web Site. Summary of Recommended Practices. After Security for a Public Web Site (April 21, 1997).

Security for a Public Web Site also recommends that a site establish security policies that mandate appropriate security practices for network administrators and users. For example, the site may want to establish a policy that requires system and network administrators to adopt and implement several of the practices described in the *Security for a Public Web Site* booklet.

d. Preventing Network File System (NFS) Attacks

Administrators of NFS environments must choose carefully whether to use NFS in a particular environment and should read all available vendor security bulletins and CERT advisories regarding NFS. (Gallen and Sutterfield, November 1996)

To remove common and known security vulnerabilities in NFS, see the NFS section of the UNIX Computer Security Checklist (December 19, 1995).

e. Preventing Network Information Service (NIS) Attacks

Garfinkel and Spafford (1996) suggest the following to prevent unauthorized disclosure of a site's NIS databases:

1. Protect the site with a firewall or at least a smart router, and not allow the UDP packets associated with RPC to cross between the site's internal network and the outside world. Because RPC uses a portmapper, i.e., portmap or rpcbind, one cannot know the actual UDP port used by RPC. The only safe strategy is to block all UDP packets except those one specifically wishes to let cross.
   Use Wietse Venema's free portmapper program, which allows one to specify a list of computers by hostname or IP address to allow or deny access to specific RPC servers. These portmapper programs are available from ftp://ftp.win.tue.nl/pub/security/portmap_4.tar.gz, and ftp://ftp.win.tue.nl/pub/security/rpcbind_2.tar.gz.
2. Some versions of NIS support the use of the /var/yp/securenets file for NIS servers. This file, when present, can specify a list of networks that may receive NIS information
3. Do not tighten up NIS but forget DNS! If the site decides that outsiders should be able to learn about the site's IP addresses, be sure to run two nameserver, one for internal use and one for external use.

f. Preventing X Window System Attacks

The best way of preventing X attacks is to properly protect X and stay current with vendor updates to X. The X Window System section of Garfinkel and Spafford (1996) suggests ways at protecting X.

4. Preventing And Detecting SATAN Attacks

TCP/IP wrapper is available through anonymous ftp from ftp://ftp.win.tue.nl/pub/security or ftp://info.cert.org/pub/tools/tcp_wrappers/.

One of the best methods in preventing a SATAN attack is to use SATAN on one's own system. Correcting the vulnerabilities that SATAN finds will prevent intruders from exploiting these vulnerabilities.

If SATAN scans one's system, programs like TCP/IP wrapper and Courtney can help to detect the scan. An installed TCP/IP wrapper program allows the host system to provide additional network logging information and gives a system administrator the ability to allow or deny access from certain systems or domains (List of Security Tools, August 1996). Courtney is a program specifically developed to detect if one's system is scanned with SATAN.

5. Preventing Trust Attacks

Two methods used to prevent spoofing attacks are not using any trust, and using cryptographic methods. The first method of not using any trust will make a site less vulnerable to host spoofing, but is perhaps the most impractical of the two methods. The second method is to use cryptographic protocols.

Although "broken" by cryptographic methods, using secure RPC protocol will provide better assurance than RPC authentication schemes that do not use any form of encryption. Examples of secure RPC protocols include secure NFS, and NIS+. Other hardware and software cryptographic solutions are being developed, but are either incomplete or require changes to system software. Smartcards are an example of a hardware solution, while Kerberos is an example of a software solution.

IV. INTERNET INTRUSION PREVENTION AND DETECTION TOOL REQUIREMENTS

ID	Requirement
IPT 1.	The tool shall detect all known intruding methods initiated over a network. For example, the tool should be able to detect if a host is being scanned with SATAN or Internet Security Scanner (ISS).
IPT 2.	The tool shall capture and analyze network traffic in and out of the a site. This requirement determines if an intruder has penetrated the site or if secret or classified data is leaving the site.
IPT 2.1.	The tool shall analyze the network traffic for all intruder methods to determine exploited vulnerabilities. This requirement determines if an intruder has penetrated the site using common tools such as SATAN and ISS.
IPT 3.	The tool shall generate information in real-time. The real-time requirement is to provide security staff and security tools the ability to react to intrusion events in real-time.
IPT 4.	The tool shall send out alarms once it detects an intruder. This requirement provides security staff with an intruder alert mechanism.
IPT 5.	The tool shall minimize false intruder alarms. An intrusion detection system shall try to minimize false alarms to minimize the cost of security personnel reacting to the alarm. Wasted effort in analyzing and determining the proper actions to stop an intrusion is a result of false alarms. This wasted effort translates to added computer security costs.
IPT 6.	The tool shall dynamically restrict intruder access to a site or host. Upon detecting an intruder, the tool shall dynamically prevent the intruder from further access. It also tries to isolate the intruder from causing any damage to the site or host.
IPT 7.	The tool shall educate operators concerning proper security procedures. The tool shall orient new operators to a site's security policies and

ID	Requirement
	procedures, and provide points of contact for not only computer security issues but all security issues.
IPT 7.1.	The tool shall educate general users in choosing safe passwords. Exploiting weak passwords is one of two methods most successful for penetrating a computer system. The other most successful method is social engineering. The tool will inform users concerning weak passwords and how to pick good passwords.
IPT 7.2.	The tool shall interact with general users in choosing safe passwords. The user will interact with the tool in choosing a password. The tool will test if the user's password is a weak or good password.
IPT 7.3.	The tool shall test a user's comprehension of the site's security procedures. The test will determine if a user understands the site's security procedures. Once a user passes the test, the site may now issue the user a computer account on the site's computer system.
IPT 8.	The tool shall educate system administrators on how to properly configure a host. The are several services that the system provides for a variety of users. The tool will instruct a system administrator on how to properly configure these services.
IPT 8.1.	The tool shall instruct a system administrator how to properly configure FTP.
IPT 8.2.	The tool shall instruct a system administrator how to properly configure SMTP.
IPT 8.3.	The tool shall instruct a system administrator how to properly configure WWW.
IPT 8.4.	The tool shall instruct a system administrator how to properly configure TFTP.
IPT 8.5.	The tool shall instruct a system administrator how to properly configure NFS.
IPT 8.6.	The tool shall instruct a system administrator how to properly configure NIS.
IPT 9.	The tool shall instruct a system administrator how to install and monitor a host's security tools.

ID	Requirement
IPT 9.1.	The tool shall instruct a system administrator how to install and monitor TCP wrappers.
IPT 10.	The tool shall provide an integrated environment for testing vulnerabilities of a host system as well as vulnerabilities of network host systems. The tool will integrate popular computer security tools, available at little or no cost, into an integrated environment. Computer security tools such as SATAN, ISS, COPS, and Crack are good candidates for integrating into a single environment.

V. DESIGN AND IMPLEMENTATION OF A INTERNET PREVENTION TOOL BY EDUCATING USERS

The focus of the thesis project is to provide a tool for educating common multi-user system users about computer security. The requirements for an intrusion detection tool given in the previous section will be a reference point for building a computer security tutorial.

The thesis project will focus on educating users on the two major successful methods for penetrating a computer system, weak passwords and social engineering. The tool will teach users how to pick good passwords and what steps to take to prevent social engineering attacks. At the end of the tutorial a test for user comprehension would follow.

The author develops this thesis project with Web based components, e.g., a Web browser, and JAVA.

A. DESIGN OF THE WEAK PASSWORD AND SOCIAL ENGINEERING PREVENTION TOOL

The weak password and social engineering prevention tool will consist of two parts. The first part of the tool is a tutorial introducing the subject of weak passwords and social engineering. The second part of the tool consists of a multiple choice test for testing the comprehension of the user.

The test portion of the tool would ideally consist of twenty-five multiple choice questions. Using multiple choice questions will make it possible to automate correcting of the test. This automating of the test will allow the user quick feedback on how well one scored on the test.

Appendix A explains the assembly of the tool. Appendix B and Appendix C contain the HTML listing of the tutorial and the testing tool driver, respectively. Appendix D contains the Java source code listing of the test.

Table 5-1 below includes possible multiple choice questions and the correct answers.

ID	Possible Test Questions	Answers
Q1.	True or false. The definition of computer security is the tools, policies, procedures, and protocols used to protect information stored and processed on a computer system. A. True B. False	A
Q2.	What are the two most successful methods used to intrude on a computer system's security? I. Social Engineering II. Exploiting weak passwords III. Sharing passwords IV. Leaving computer logged into one's account V. Exploiting an operating system vulnerability A. I & II B. II & III C. IV & V D. I & III E. II & V	A

ID	Possible Test Questions	Answers
Q3.	Select the best answer from A through E. What attributes make up a good password? I. Easy to remember II. Contain acronyms common to a specific field or profession III. Difficult to guess IV. Have digits and punctuation characters as well as letters V. Are a variation of a first or a last name A. I, II, & IV B. I, III, & IV C. I, IV, & V D. None of the above E. All of the above	B
Q4.	Select the best answer from A through E. What attributes make up weak password? I. Have both uppercase and lowercase letters II. Are in a dictionary of some language III. Contain no password at all IV. Combine short words with a special character or a number V. Contain a proper noun, e.g., the name of a real or fictitious character or place A. I, IV, & V B. I, II, & IV C. II, III, & V D. None of the above E. All of the above	C

ID	Possible Test Questions	Answers
Q5.	True or false. Social Engineering is the process of using social interactions to obtain information about a company's computer system? A. True B. False	A
Q6.	Select the best answer from A through E. Social Engineering uses which of the following methods? I. Impersonating system or security administration staff for information gathering II. Blackmail III. Physical threat IV. Rummaging through a company's garbage V. Become an employee of the company for the purpose of gathering information A. I & II B. II & III C. I, IV, & V D. None of the above E. All of the above	E
Q7.	Is it proper to share passwords? A. Generally yes B. Generally no C. Yes D. No E. Depends on the site's policy	D

Table 5-1 Possible Test Questions

VI. CONCLUSION AND FUTURE RESEARCH

Computer and network security are challenges that are ever changing and getting more complex as time goes on. System and security administrators are overburdened dealing with the current methods of intrusions and vulnerabilities of their systems. Administrators also look for new and innovative ways to guard against new methods and vulnerabilities. The use of cryptography and potentially expensive technical means will enhance the security of one's systems. This security will fail when one neglects security education of users.

The thesis addresses a portion of the security education problem by designing and developing a tool to educate users on the two major successful methods for penetrating a computer system, weak passwords and social engineering. This prevention tool explains what attributes make up a good password and what attributes make up a weak password. The tool also provides guidelines for choosing good passwords, and for protecting passwords. The tool consists of a tutorial and end with an exam to test user comprehension concerning picking good passwords and preventing social engineering attacks.

The development of the weak password and social engineering tool is an attempt, in a small way, to help administrators maintain security in computer systems and networks.

Future enhancements to the weak password and social engineering prevention tool may include an interactive aid for picking passwords. An interactive password aid would assist the user in selecting good passwords. This password aid may integrate a password cracking utility like Crack or Jack the Ripper. On the weak password portion of the tutorial, Crack and Jack the Ripper are two examples of UNIX password cracking software available free on the Internet.

Enhancement to the social engineering portion of a tutorial may involve the user in a simulated social engineering attack. This simulation would present ways for the user to deal with such an attack.

Future tools in the area of user education could include a tool for orienting new system administrators on computer security duties. Such a tool could consist of an integrated toolkit or control panel of various security tools and an interactive tutorial in the use of the integrated toolkit. Candidate security tools for this integrated toolkit are Security Analysis Tool for Auditing Networks (SATAN), Internet Security Scanner (ISS), Crack, and Computer Oracle and Password System (COPS).

SATAN and a stripped-down version of ISS are two network scanners available free on the Internet. A more complex version of ISS is commercially available. The use of these scanners may help determine vulnerabilities in one's network that need fixing. Unfortunately, intruders can also use these tools to find network vulnerabilities to exploit. (Garfinkel and Spafford, 1996)

As mentioned above, Crack is a Unix password cracking utility. Note that similar password cracking software is available on Windows NT. COPS is a collection of short shell scripts and C programs that perform vulnerability checks of one's computer system. Crack and COPS are available free on the Internet. (Garfinkel and Spafford, 1996)

LIST OF REFERENCES

A Guide to Developing Computing Policy Documents, USENIX Association, 1996.

"Anonymous FTP Abuses", CERT(sm) Coordination Center, 1996, ftp://info.cert.org/pub/tech_tips/anonymous_ftp_abuses.

"ANONYMOUS FTP CONFIGURATION GUIDELINES", CERT(sm) Coordination Center, 1995, ftp://info.cert.org/pub/tech_tips/anonymous_ftp_config.

"List of Security Tools", CERT(sm) Coordination Center, Version 1.1, August 1996, ftp://info.cert.org/pub/tech_tips/security_tools.

"Net Perceptions", InformationWeek, iss. 629, May 5, 1997, p. 14.

"UNIX Computer Security Checklist" The Australian Computer Emergency Response Team, Version 1.1, December 19, 1995, ftp://info.cert.org/pub/tech_tips/AUSCERT_checklist1.1.

"UNIX Configuration Guidelines", CERT(sm) Coordination Center, Version 1.1, August 1996, ftp://info.cert.org/pub/tech_tips/UNIX_configuration_guidelines.

Aleph One <aleph1@DFW.NET>, "Buffer Overflows: A Summary", Email to Bugtraq List <BUGRAQ@NETSPACE.ORG>, Wed, 30 Apr 1997 01:03:17 -0500.

Aleph One <aleph1@underground.org>, "Smashing The Stack For Fun And Profit", Phrack 49, v. 7, no. 49, November 8, 1996, 14 of 16.

Anderson, Ross J., "Why Cryptosystems Fail", Communications of the ACM, vol. 37, no. 11, November 1994, pp. 32-40.

Farmer, Dan, and Venema, Wietse, "Improving the Security of Your Site by Breaking Into It", Posted to Usenet, December 2, 1993, ftp://ftp.win.tue.nl/pub/security/admin-guide-to-cracking.101.Z.

Gallen, Bob and Sutterfield, Lee, "Network Security Points of Failure", UNIX Review, vol. 14, no. 12, November 1996, pp. 47-53.

Garfinkel, Simson and Spafford, Gene, *Practical Unix & Internet Security*, O'Reilly & Associates, Inc., 1996.

Instenes, Shawn, "Stack Smashing: What To Do?", ;login, vol. 22, no. 2, April 1997, p. 18.

Lundy, Bert, "Tutorial on Network Security", PSTV/Forte Conference, November 1997, Osaka, Japan.

Oppenheimer, David L., Wagner, David A., and Crabb, Michele D, *Systems Security: A Management Perspective*, USENIX Association, 1997.

Russell, Deborah and Gangemi Sr., G. T., *Computer Security Basics*, O'Reilly & Associates, Inc., 1991.

Schneier, Bruce, *Applied Cryptography Second Edition: Protocols, Algorithms, and Source Code in C*, John Wiley & Sons, Inc., 1996.

Security for a Public Web Site, Security Improvement Team Networked Systems Survivability Program, Software Engineering Institute & Carnegie Mellon University, April 21, 1997.

Smith, Nathan P., "Stack Smashing Vulnerabilities in the UNIX Operating System", Computer Science Department Southern Connecticut State University, http://millcomm.com/~nate/machines/security/stack-smashing/, May 7, 1997.

Stallings, William, *Network and Internetwork Security: Principles and Practice*, Prentice Hall, Inc., 1995.

Stoll, Clifford, The Cuckoo's Egg, Doubleday, 1989.

Thyfault, Mary E. and Davis, Beth, "Your Wide Area Internet", InformationWeek, iss. 629, May 5, 1997, pp. 14,15.

Webster's New World Dictionary of Computer Terms, Simon & Schuster Inc., 1993

Webster's Ninth New Colletiate Dictionary, Merriam-Webster Inc., 1991.

Winkler, Ira S. and Dealy, Brian, "Information Security Technology?...Don't Rely on It A Case Study in Social Engineering", Proceedings of the Fifth USENIX UNIX Security Symposium, June 1995, Salt Lake City, Utah.

BIBLIOGRAPHY

"Choosing an Operating System", CERT(sm) Coordination Center, July 23, 1996, ftp://info.cert.org/pub/tech_tips/choose_operating_sys.

"Email Bombing and Spamming", CERT(sm) Coordination Center, 1996, ftp://info.cert.org/pub/tech_tips/email_bombing_spamming.

"Intruder Detection Checklist", CERT(sm) Coordination Center, Version 1.1, August 1996, ftp://info.cert.org/pub/tech_tips/intruder_detection_checklist.

"Packet Filtering for Firewall Systems", CERT(sm) Coordination Center, February 1996, ftp://info.cert.org/pub/tech_tips/packet_filtering.

"Protecting Yourself from Password File Attacks", CERT(sm) Coordination Center, 1996, ftp://info.cert.org/pub/tech_tips/passwd_file_protection.

"Spoofed/Forged Email", CERT(sm) Coordination Center, 1995, ftp://info.cert.org/pub/tech_tips/email_spoofing.

"Steps for Recovering from a UNIX Root Compromise", CERT(sm) Coordination Center, Version 3.1, August 1996, ftp://info.cert.org/pub/tech_tips/root_compromise.

alhambra <alhambra@infonexus.com>, "Network Management Protocol Insecurity: SNMPv1", Phrack 50, v. 7, no. 50, April 9, 1997, 7 of 16.

Anderson, Ross J., "A Security Policy Model for Clinical Information Systems", Proceedings 1996 IEEE Symposium on Security and Privacy, May 6-8, 1996, Oakland, CA, p. 30-43.

Barnett, Bruce, and Vu, Dai N., "Vulnerability Assessment and Intrusion Detection with Dynamic Software Agents", Software Technology Conference, Track 9 Security, April 29, 1997, Salt Lake City, UT.

Barnett, Stephen F., "Computer Security Training and Education: A Needs Analysis", Proceedings 1996 IEEE Symposium on Security and Privacy, May 6-8, 1996, Oakland, CA, p. 26, 27.

Browne, Randy, "Extended Abstract: An Architecture for Covert Channel Control in RealTime Networks and MultiProcessors", Proceedings 1995 IEEE Symposium on Security and Privacy, May 8-10, 1995, Oakland, CA, p. 155-168.

Cooper, David A., and Birman, Kenneth P., "Preserving Privacy in a Network of Mobile Computers", Proceedings 1995 IEEE Symposium on Security and Privacy, May 8-10, 1995, Oakland, CA, p. 26-38.

Cove, David J., "Collaring the Cybercrook: An Investigator's View", IEEE Spectrum, v. 34, no. 6, June 1997, pp. 31-36.

D'haeseleer, Patrik, Forrest, Stephanie, and Helman, Paul, "An Immunological Approach to Change Detection: Algorithms, Analysis and Implications", Proceedings 1996 IEEE Symposium on Security and Privacy, May 6-8, 1996, Oakland, CA, p. 110-119.

daemon9 AKA route <route@infonexus.com> and alhambra <alhambra@infonexus.com>, "Project Loki: ICMP Tunneling", Phrack 49, v. 7, no. 49, November 8, 1996, 6 of 16.

daemon9 AKA route <route@infonexus.com>, "Project Hades: TCP weaknesses", Phrack 49, v. 7, no. 49, November 8, 1996, 7 of 16.

Dean, Drew, Felten, Edward W., and Wallach, Dan S., "Java Security: From HotJava to Netscape and Beyond", Proceedings 1996 IEEE Symposium on Security and Privacy, May 6-8, 1996, Oakland, CA, p. 190-200.

Deng, Robert H., Bhonsle, Shailendra K., Wang, Weiguo, and Lazar, Aurel A., "Integrating Security in CORBA Based Object Architectures", Proceedings 1995 IEEE Symposium on Security and Privacy, May 8-10, 1995, Oakland, CA, p. 50-61.

Detecting Signs of Intrusion, Security Improvement Team Networked Systems Survivability Program, Software Engineering Institute & Carnegie Mellon University, April 21, 1997.

Doorn, Leendert van, Abadi, Martin, Burrows, Mike, and Wobber, Edward, "Secure Network Objects", Proceedings 1996 IEEE Symposium on Security and Privacy, May 6-8, 1996, Oakland, CA, p. 211-221.

Forrest, Stephanie, and Longstaff, Thomas A., "A Sense of Self for Unix Processes", Proceedings 1996 IEEE Symposium on Security and Privacy, May 6-8, 1996, Oakland, CA, p. 120-128.

Gilliss, Gregory, "CGI Security Holes", Phrack 49, v. 7, no. 49, November 8, 1996, 8 of 16.

Irvine, Cynthia E., "Goals for Computer Security Education", Proceedings 1996 IEEE Symposium on Security and Privacy, May 6-8, 1996, Oakland, CA, p. 24, 25.

Kang, Myong H, Moskowitz, Ira S., and Lee, Daniel C., "A Network Version of The Pump", Proceedings 1995 IEEE Symposium on Security and Privacy, May 8-10, 1995, Oakland, CA, p. 144-154.

Maimon, Uriel, "Port Scanning without the SYN flag", Phrack 49, v. 7, no. 49, November 8, 1996, 15 of 16.

Mao, Wenbo, "On two Proposals for On-line Bankcard Payments using Open Networks: Problems and Solutions", Proceedings 1996 IEEE Symposium on Security and Privacy, May 6-8, 1996, Oakland, CA, p. 201-210.

nate <nate@MILLCOMM.COM>, "Smashing the Stack: prevention?", Email to Bugtraq List <BUGRAQ@NETSPACE.ORG>, Sun, 27 Apr 1997 20:31:55 -0400.

Needham, Roger M., "Denial of Service; An Example", Communications of the ACM, vol. 37, no. 11, November 1994, pp. 42-46.

Report of the Defense Science Board Task Force on Information Warfare - Defense, Office of the Under Secretary of Defense for Acquisition & Technology, November 1996.

Staniford-Chen, Stuart, "Holding Intruders Accountable on the Internet", Proceedings 1995 IEEE Symposium on Security and Privacy, May 8-10, 1995, Oakland, CA, p. 39-49.

Venema, Wietse, "Murphy's law and computer security", Proceedings of the Sixth USENIX UNIX Security Symposium, July 1996, San Jose, CA.

Venkatraman, Balaji R., and Newman-Wolfe, R. E., "Capacity Estimation and Auditability of Network Covert Channels", Proceedings 1995 IEEE Symposium on Security and Privacy, May 8-10, 1995, Oakland, CA, p. 186-198.

Woo, Thomas Y. C., and Lam, Simon S., "Authentication for Distributed Systems", To appear in Internet Security, Dorothy Denning and Peter Denning (eds.), Addison-Wesley and ACM Press Books, ftp://ftp.cs.utexas.edu/pub/lam/denning.ps.Z.

Young, Adam, and Yung, Moti, "Cryptovirology: Extortion-Based Security Threats and Countermeasures", Proceedings 1996 IEEE Symposium on Security and Privacy, May 6-8, 1996, Oakland, CA, p. 129-140.

Zerkle, Dan, and Levitt, Karl, "NetKuang–A Multi-Host Configuration Vulnerability Checker", Proceedings of the Sixth USENIX UNIX Security Symposium, July 1996, San Jose, CA.

GLOSSARY

Computer & Network Security The tools, policies, procedures, and protocols used to protect information while being stored and processed on a computer system, and while being transmitted through a network.

Computer Security The tools, policies procedures, and protocols used to protect information while being stored and processed on a computer system.

Detect To discover the true character of, or to discover or determine the existence, presence, or fact of.

Detection The act of detecting or the state or fact of being detected.

Espionage The practice of spying or the use of spies to obtain information about the plans and activities especially of a foreign government or a competing company.

Intrude To thrust or force in or upon especially without permission, welcome, or fitness. To cause to enter as if by force.

Intruder An intruder is an entity that makes unauthorized access to a computer system or network. This entity may be a person, an automated agent, or a combination of the two.

Intrusion The act of intruding or the state of being intruded; especially the act of wrongfully entering upon, seizing, or taking possession of the property of another.

LAN Local Area Network

Network Security The tools, policies, procedures, and protocols used to protect information while being transmitted through a network.

Prevent To keep from happening or existing.

Prevention The act of preventing or hindering.

Privacy Freedom from unauthorized disclosure.

Protect To cover or shield from exposure, injury, or destruction.

Protection The act of protecting or the state of being protected.

Secrecy The habit or practice of keeping secrets or maintaining privacy or concealment.

Secret Something kept from the knowledge of others or shared only confidentially with a few.

Secure To relieve from exposure to danger.

Security Measures taken to guard against espionage or sabotage, crime, attack, or escape.

Social Engineering Process of using social interactions to obtain information about a "victim's" computer system.

WAN Wide Area Network

APPENDIX A. TOOL ASSEMBLY

The following components make up the weak password and social engineering prevention tool:

1. prevention_tool.html
2. TestTool.html
3. TestTool.java
4. TestTool.class
5. TestToolFrame.class

The file prevention_tool.html contains the tutorial portion of the tool prevention, which is in Hypertext Markup Language (HTML) format. TestTool.html file contains the HTML driver used to integrate the tutorial portion with the testing portion of the tool. The file TestTool.java is the Java source code for the testing portion of the tool. When compiled, the TestTool.java source produces the TestTool.class and TestToolFrame.class class files.

The implementation of this prevention tool took place on a Power Macintosh 8500/120 with 32 MB RAM, running Mac OS 7.5.5. The tutorial produced by using rtftohtml version 3.8 to convert a Rich Text Format (RTF) of the tutorial to HTML. Chris Hector is the author of the rtftohtml application. The Java source compiled using the Java Development Kit (JDK) 1.0.2 from Sun Microsystems, Inc.

To use the prevention tool one must use a Java aware browser to open the file prevention_tool.html. Examples of Java aware browsers are Netscape Navigator™ or Sun's HotJava.

Appendix B and Appendix C contain the HTML listing of prevention_tool.html and TestTool.html, respectively. Appendix D contains the Java source code listing of TestTool.java.

APPENDIX B. PREVENTION TOOL TUTORIAL HTML LISTING

```
<HTML><HEAD>
<!-- This document was created from RTF source by rtftohtml version 3.8 -->
<TITLE>Development of A Internet Intrusion Detection Tool</TITLE></HEAD>

<p><center><b>DEVELOPMENT OF AN INTERNET INTRUSION PREVENTION
TOOL</b></center><br>
<center><b>Dagohoy Hofilena Anunciado</b></center><br>
<center><b>B.S. in Computer Science, University of California, San Diego - June
1990</b></center><br>
<center><b>Master of Science in Software Engineering - December
1997</b></center><br>
<center><b>Thesis Advisor: Bert Lundy, Department of Computer Science, Naval
Postgraduate School<br>Co-Advisor: Ron Broersma, SPAWARSYSCEN San
Diego</b></center><br>
The purpose of the Internet Intrusion Prevention Tool is to educate users on
the two major successful methods for penetrating a computer system, weak
passwords and social engineering. The tool will teach users how to pick good
passwords and what steps to take to prevent social engineering attacks. The
tool will consist of a tutorial and end with an exam to test user comprehension
concerning picking good passwords and preventing social engineering attacks.<br>
<br>
<b>KEYWORDS: </b>Computer Security, Social Engineering, Intrusion Prevention<br>
<br>
<b>DOD KEY TECHNOLOGY AREA:</b> Computing and Software<br>
<br>
<hr size=4>

<H1>
Development of A Internet Intrusion Detection Tool</H1>

<center><b>TABLE OF CONTENTS</b></center>
<BODY BACKGROUND="" BGCOLOR="#FFFFFF" TEXT="#000000">
<UL>
<LI><A HREF="#Heading1">I. INTRODUCTION</A>
<LI><A HREF="#Heading2">II. COMPUTER AND NETWORK SECURITY</A>
<UL>
<LI><A HREF="#Heading3">A. DEFINITION OF AN INTRUDER</A>
<LI><A HREF="#Heading4">B. METHODS OF IDENTIFICATION</A>
<UL>
<LI><A HREF="#Heading5">1. Authentication Using Passwords</A>
<UL>
<LI><A HREF="#Heading6">a) Difficult to Guess Passwords</A>
<LI><A HREF="#Heading7">b) Weak Passwords</A>
<LI><A HREF="#Heading8">c) Good Password</A>
<LI><A HREF="#Heading9">d) Protecting Passwords</A>
</UL>
</UL>
<LI><A HREF="#Heading10">C. EFFECTIVE METHODS USED TO INTRUDE</A>
<UL>
<LI><A HREF="#Heading11">1. Social Engineering</A>
<LI><A HREF="#Heading12">2. Password Cracking</A>
</UL>
<LI><A HREF="#Heading13">D. PREVENTING SOCIAL ENGINEERING</A>
<UL>
<LI><A HREF="#Heading14">1. Do Not Rely Upon Common Internal Identifiers</A>
```

```
<LI><A HREF="#Heading15">2. Implement a Call Back Procedure When Disclosing Protected
Information</A>
<LI><A HREF="#Heading16">3. Implement a Security Awareness Program</A>
<LI><A HREF="#Heading17">4. Identify Direct Computer Support Analysts</A>
<LI><A HREF="#Heading18">5. Create A Security Alert System</A>
<LI><A HREF="#Heading19">6. Social Engineering to Test Security Policies</A>
</UL>
<LI><A HREF="#Heading20">E. PREVENTING PASSWORD CRACKING</A>
</UL>
<LI><A HREF="TestTool.html">III. TAKE TEST</A>
</UL>
<hr size=4>
<a name="_Toc336357867"></a><a name="_Toc336358437"></a><a name="_Toc349494303"></a><a
name="_Toc389306842"></a><a name="_Toc389306965"></a><a name="_Toc389307309"></a><a
name="_Toc389307861"></a><a name="_Toc389309205"></a><a name="_Toc389309268"></a><a
name="_Toc389309535"></a><a name="_Toc389311481"></a><a name="_Toc389311612"></a><a
name="_Toc389311908"></a><a name="_Toc389312137"></a><a name="_Toc389312521"></a><a
name="_Toc389312900"></a><a name="_Toc389313306"></a><a name="_Toc390006731"></a><a
name="_Toc393704944"></a><a name="_Toc393708820"></a><a name="_Toc395095743"></a><a
name="_Toc395275777"></a><a name="_Toc396448538"></a><a name="_Toc396532986"></a><a
name="_Toc396811698"></a><a name="_Toc397381598"></a><a name="_Toc397382154"></a><a
name="_Toc398069412"></a><a name="_Toc398069597"></a><a name="_Toc398075246"></a><a
name="_Toc401601881"></a><a name="_Toc402709899"></a><a name="_Toc402888079"></a><a
name="_Toc403222560"></a><a name="_Toc403316797"></a><a name="_Toc403805128"></a><a
name="_Toc404008187"></a><a name="_Toc404342507"></a><a name="_Toc404342559"></a><a
name="_Toc404437781"></a><a name="_Toc404438866"></a><a name="_Ref404527591"></a><a
name="_Toc406789604"></a></p>

<h1>
<A NAME="Heading1">I. INTRODUCTION</A></h1>

<p>Many
of the world's businesses today are moving toward the use of the Internet. Many
companies see the Internet playing a more fundamental role in their operations,
as an inexpensive, easy-to-deploy wide area transport network. Fifty-one
Fortune 1000 companies were asked if the Internet could replace other Wide Area
Network (WAN) technologies. The response was 45% yes, 6% already happening, 27%
no, and 22% maybe. In a follow-up question, "What capabilities would the
Internet need before you would use it as a WAN?" The top response of 75% of
companies was the need for better security.<br>
<br>
To provide better security to computer systems and the Internet will require
educating the users and maintainers of these services. The solution to this
problem of providing better security requires that all computer users get
involved. Users need to realize that attacks on computer systems and the
Internet are not just attacks on individual systems, but are attacks on the
entire community. Overburdened maintainers need automated tools to help secure
these services. There is a need to develop an infrastructure that has security
built into it from the beginning.<br>
<br>
The tutorial addresses a portion of security education by educating users on
the two major successful methods for penetrating a computer system or network,
weak passwords and social engineering. The tutorial teaches users how to pick
good passwords and what steps to take to prevent social engineering attacks.
The tutorial consists of a tutorial and end with an exam to test user
comprehension concerning picking good passwords and preventing social
engineering attacks.
```

```
<a name="_Toc389306843"></a><a name="_Toc389306966"></a><a name="_Toc389307310"></a><a
name="_Toc389307862"></a><a name="_Toc389309206"></a><a name="_Toc389309269"></a><a
name="_Toc389309536"></a><a name="_Toc389311482"></a><a name="_Toc389311613"></a><a
name="_Toc389311909"></a><a name="_Toc389312138"></a><a name="_Toc389312522"></a><a
name="_Toc389312901"></a><a name="_Toc389313307"></a><a name="_Toc390006732"></a><a
name="_Toc393704949"></a><a name="_Toc393708825"></a><a name="_Toc395095748"></a><a
name="_Toc395275782"></a><a name="_Toc396448543"></a><a name="_Toc396532991"></a><a
name="_Toc396811702"></a><a name="_Toc397381602"></a><a name="_Toc397382158"></a><a
name="_Toc389306844"></a><a name="_Toc389306967"></a><a name="_Toc389307311"></a><a
name="_Toc389307863"></a><a name="_Toc389309207"></a><a name="_Toc389309270"></a><a
name="_Toc389309537"></a><a name="_Toc389311483"></a><a name="_Toc389311614"></a><a
name="_Toc389311910"></a><a name="_Toc389312139"></a><a name="_Toc389312523"></a><a
name="_Toc389312902"></a><a name="_Toc389313308"></a><a name="_Toc390006733"></a><a
name="_Toc393704950"></a><a name="_Toc393708826"></a><a name="_Toc395095749"></a><a
name="_Toc395275783"></a><a name="_Toc396448544"></a><a name="_Toc396532992"></a><a
name="_Toc396811703"></a><a name="_Toc397381603"></a><a name="_Toc397382159"></a><a
name="_Toc398069417"></a><a name="_Toc398069602"></a><a name="_Toc398075251"></a><a
name="_Ref400143199"></a><a name="_Ref400314954"></a><a name="_Ref400314991"></a><a
name="_Ref400315047"></a><a name="_Toc401601886"></a><a name="_Toc402709904"></a><a
name="_Toc402888084"></a><a name="_Toc403222565"></a><a name="_Toc403316802"></a><a
name="_Toc403805133"></a><a name="_Toc404008192"></a><a name="_Toc404342509"></a><a
name="_Toc404342561"></a><a name="_Toc404437783"></a><a name="_Toc404438868"></a><a
name="_Ref404527283"></a><a name="_Ref404539978"></a><a name="_Ref404540852"></a><a
name="_Toc406789605"></a></p>

<h1>
<A NAME="Heading2">II. COMPUTER AND NETWORK SECURITY</A></h1>

<a name="_Toc393704955"></a><a name="_Toc393708831"></a><a name="_Toc395095754"></a><a
name="_Toc395275788"></a><a name="_Toc396448549"></a><a name="_Toc396532997"></a><a
name="_Toc396811708"></a><a name="_Toc397381608"></a><a name="_Toc397382164"></a><a
name="_Toc398069422"></a><a name="_Toc398069607"></a><a name="_Toc398075260"></a><a
name="_Toc401601895"></a><a name="_Toc402709913"></a><a name="_Toc402888093"></a><a
name="_Toc403222574"></a><a name="_Toc403316811"></a><a name="_Toc403805142"></a><a
name="_Toc404008201"></a><a name="_Toc404342518"></a><a name="_Toc404342570"></a><a
name="_Toc404437792"></a><a name="_Toc404438877"></a><p>For
the purposes of this tutorial the definition of <i>computer and network
security</i> are the tools, policies, procedures, and protocols used to protect
information being stored and processed on a computer system, and while being
transmitted through a network.<br>
<br>
Primary concern of computer security is with the protection of individual
computer systems, whether or not the system is connected to a network. Primary
concern of network security is with communication channels. Many of the threats
to computers enter from a network channel and the endpoints of communications
channels are usually computers. One can not ignore either of the concerns of
computer security or network security to have a comprehensive computer and
network security.
<a name="_Toc406789606"></a></p>

<h2>
<A NAME="Heading3">A. DEFINITION OF AN INTRUDER</A></h2>

<p>An
intruder is an entity that makes unauthorized access to a computer system or
network. This entity may be a person, an automated agent, or a combination of
the two.
<a name="_Toc406789607"></a></p>
```

```
<h2>
<A NAME="Heading4">B. METHODS OF IDENTIFICATION</A></h2>

<p>The
one way in which a computer system provides security is by controlling access
to that system. This access is usually controlled by forcing one to identify
oneself and having the system authenticate one's identity. In a multi-user
system there are three classic methods for proving one's identity.<br>
<br>
The first method is knowing something that no one else knows. For example,
having a secret password to an account that no one else knows implies ownership
of the account. In the majority of cases this method provides a more than
adequate means of proving one's identity. The weakness of this method is
passwords may be stolen or obtained by other means. A stolen password may have
come about by writing the password down and someone else reading that password.
One may have given the password away. The password may be easy to guess, or
found through systematic cracking methods.<br>
<br>
The second method for proving one's identity is to have something that no one
else has. This may be a key, token, badge, or smart card. To have something
that no one else has implies ownship of the account. The weakness with this
method of proving one's identity is a key or equivalent may be stolen, lost, or
duplicated.<br>
<br>
The third and final classic method used to prove identity is by biometrics.
This method uses a previously stored biometric to prove one's identity.
Biometrics is the use of physical or behavior traits to identify a person.
These traits may include fingerprint, handprint, retina pattern, voice,
signature, or keystroke pattern. Biometrics systems are quite accurate, yet on
occasions, reject valid users and accept invalid ones. The problem with this
system is that many people are not comfortable using it.<br>
<br>
An additional method used to prove one's identity is to used a combination of
the above methods. By using a combination of methods, the weaknesses of an
individual method are minimized. For example, using a combination of a password
and smart card to prove one's identity would still prevent an intruder who
somehow obtained the password from accessing the system, and visa versa.
<a name="_Ref404540330"></a><a name="_Toc406789608"></a></p>

<h3>
<A NAME="Heading5">1. Authentication Using Passwords</A></h3>

<p>The
use of passwords is still the authentication tool of choice, with the use of
smart cards and biometrics as secondary authentication tools. Most systems
require one to identify themselves with a login identifier followed by a
password. UNIX and Windows NT systems are example systems.
<a name="_Ref404542123"></a></p>

<h4>
<A NAME="Heading6">a) Difficult to Guess Passwords</A></h4>

<p>Difficult
to guess passwords are the main defense against intruders. The best passwords
are difficult to guess because they:</p>
<p>1.   Have both uppercase and lowercase letters.<br>
2.      Have digits and punctuation characters as well as letters. Be cautious with
```

```
some special characters, e.g., #, and @, for these may have special meaning to
terminal emulation software.<br>
3.      May include some control characters and spaces. Control characters like
CONTROL-S, CONTROL-H, CONTROL-/, and CONTROL-\ can cause problem with terminal
emulation software.<br>
4.      Are easy to remember, and so need not be written down.</p>

<h4>
<A NAME="Heading7">b) Weak Passwords</A></h4>

<p>Easy
to guess and weak passwords may have the following attributes:</p>
<p>1.   Are in a dictionary of some language.<br>
2.      Contain a proper noun, e.g., the name of a real or fictitious character or
place.<br>
3.      Are acronyms common to some specific field or profession.<br>
4.      Are a variation of a first or a last name.<br>
5.      Match the login identifier or account.<br>
6.      Contain the vender-supplied default passwords.<br>
7.      Contain no password at all.
<a name="_Ref404541129"></a></p>

<h4>
<A NAME="Heading8">c) Good Password</A></h4>

<p>The
following are some suggestions for picking a good password:</p>
<p>1.   Combine short words with a special character or a number, e.g., sale2noon
or love-hate.<br>
2.      Choose an easy-to-remember phrase or lyric, and use the first letters to
form a password. Also add punctuation or mixed case letters. For example, "Hush
a my baby", would become Ham-b-.<br>
3.      Pick a nonsense word that is still pronounceable, e.g., cU2Morow or U4eyes.
<a name="_Ref404541255"></a></p>

<h4>
<A NAME="Heading9">d) Protecting Passwords</A></h4>

<p>Passwords
and the file that store passwords need protection. The following are some
suggestions for protecting passwords:</p>
<p>1.   Make sure that all logins have password.<br>
2.      Change all system, test, or guest passwords before allowing users to log in.
Example system, test, or guest accounts are root, system, test, demo, and
Administrator.<br>
3.      Do not ever share user passwords.<br>
4.      Do not write a password down, especially on a terminal, a computer, or
anywhere around a workspace. If one does write down a password, do not identify
it as a password. One may want to disguise a password by expanding it to a
phrase.<br>
5.      Do not type a password while anyone is watching.<br>
6.      Do not record a password online or send a password using electronic mail.
<i>The Cuckoo's Egg</i>, Clifford Stoll (1989), relates a story of how an
intruder gathered valid passwords by scanning a system's email and text files
for the word "password". If one does need to store passwords and accounts
online or send them using email, use encryption to protect the file or message.
<br>
7.      Change a password immediately if one believes the password is stolen.<br>
```

```
8.      Change a password on a regular basis, even if a password is not compromised.
<a name="_Toc389306845"></a><a name="_Toc389306968"></a><a name="_Toc389307312"></a><a
name="_Toc389307864"></a><a name="_Toc389309208"></a><a name="_Toc389309271"></a><a
name="_Toc389309538"></a><a name="_Toc389311484"></a><a name="_Toc389311615"></a><a
name="_Toc389311911"></a><a name="_Toc389312140"></a><a name="_Toc389312524"></a><a
name="_Toc389312903"></a><a name="_Toc389313309"></a><a name="_Toc390006734"></a><a
name="_Toc393704957"></a><a name="_Toc393708833"></a><a name="_Toc395095756"></a><a
name="_Toc395275790"></a><a name="_Toc396448551"></a><a name="_Toc396532999"></a><a
name="_Toc396811710"></a><a name="_Toc397381610"></a><a name="_Toc397382166"></a><a
name="_Toc398069424"></a><a name="_Toc398069609"></a><a name="_Toc398075262"></a><a
name="_Toc401601897"></a><a name="_Toc402709915"></a><a name="_Toc402888095"></a><a
name="_Toc403222576"></a><a name="_Toc403316813"></a><a name="_Toc403805144"></a><a
name="_Toc404008202"></a><a name="_Toc404342519"></a><a name="_Toc404342571"></a><a
name="_Toc404437793"></a><a name="_Toc404438878"></a><a name="_Toc406789609"></a></p>

<h2>
<A NAME="Heading10">C. EFFECTIVE METHODS USED TO INTRUDE</A></h2>

<p>Effective
methods used to intrude on a system's security are the use of social
engineering and password cracking.
<a name="_Toc389306846"></a><a name="_Toc389306969"></a><a name="_Toc389307313"></a><a
name="_Toc389307865"></a><a name="_Toc389309209"></a><a name="_Toc389309272"></a><a
name="_Toc389309539"></a><a name="_Toc389311485"></a><a name="_Toc389311616"></a><a
name="_Toc389311912"></a><a name="_Toc389312141"></a><a name="_Toc389312525"></a><a
name="_Toc389312904"></a><a name="_Toc389313310"></a><a name="_Toc390006735"></a><a
name="_Toc393704958"></a><a name="_Toc393708834"></a><a name="_Toc395095757"></a><a
name="_Toc395275791"></a><a name="_Toc396448552"></a><a name="_Toc396533000"></a><a
name="_Toc396811711"></a><a name="_Toc397381611"></a><a name="_Toc397382167"></a><a
name="_Toc398069425"></a><a name="_Toc398069610"></a><a name="_Toc398075263"></a><a
name="_Toc401601898"></a><a name="_Toc402709916"></a><a name="_Toc402888096"></a><a
name="_Toc403222577"></a><a name="_Toc403316814"></a><a name="_Toc403805145"></a><a
name="_Toc404008203"></a><a name="_Toc404342520"></a><a name="_Toc404342572"></a><a
name="_Toc404437794"></a><a name="_Toc404438879"></a><a name="_Toc406789610"></a></p>

<h3>
<A NAME="Heading11">1. Social Engineering</A></h3>

<p>The
hacker community associates the term Social Engineering with the process of
using social interactions to obtain information about a "victim's" computer
system. This interaction may be as simple as calling a company and asking
people for their passwords, possibly by impersonating a company's information
security staff. Blackmail and physical threat are other methods of Social
Engineering. More elaborate methods used to obtain a company's sensitive
information include going through the company's garbage, posing as an employee
of the company, or even becoming an employee of the company.
<a name="_Toc396448553"></a><a name="_Toc396533001"></a><a name="_Toc396811712"></a><a
name="_Toc397381612"></a><a name="_Toc397382168"></a><a name="_Toc398069426"></a><a
name="_Toc398069611"></a><a name="_Toc398075264"></a><a name="_Toc401601899"></a><a
name="_Toc402709917"></a><a name="_Toc402888097"></a><a name="_Toc403222578"></a><a
name="_Toc403316815"></a><a name="_Toc403805146"></a><a name="_Toc404008204"></a><a
name="_Toc404342521"></a><a name="_Toc404342573"></a><a name="_Toc404437795"></a><a
name="_Toc404438880"></a><br>
<br>
The advantages of social engineering are low cost, simple to implement, time
efficient, and high success at getting around a company's formidable security.
<a name="_Toc406789611"></a></p>
```

```
<h3>
<A NAME="Heading12">2. Password Cracking</A></h3>

<p>Password
cracking is obtaining a target host's password file and running password
cracking software against the password file. This password cracking method
attempts to find valid passwords for accounts in the password file. Examples of
UNIX password cracking software are Crack and John the Ripper. Note that
similar password cracking software is available on Windows NT. Finding the root
password would be the most threatening to a system's security, since an
intruder will have total access to the targeted host. One of the best defenses
against password cracking is having difficult-to-guess passwords for all
accounts in the password file. See Section II.1.a for guidelines in choosing
difficult to guess passwords.
<a name="_Toc395095765"></a><a name="_Toc395275800"></a><a name="_Toc396448563"></a><a
name="_Toc396533011"></a><a name="_Toc396811722"></a><a name="_Toc397381622"></a><a
name="_Toc397382178"></a><a name="_Toc398069434"></a><a name="_Toc398069619"></a><a
name="_Toc398075273"></a><a name="_Toc401601908"></a><a name="_Toc402709926"></a><a
name="_Toc402888103"></a><a name="_Toc403222584"></a><a name="_Toc403316820"></a><a
name="_Toc403805151"></a><a name="_Toc404008209"></a><a name="_Toc404342526"></a><a
name="_Toc404342578"></a><a name="_Toc404437800"></a><a name="_Toc404438885"></a><a
name="_Toc406789612"></a></p>

<h2>
<A NAME="Heading13">D. PREVENTING SOCIAL ENGINEERING</A></h2>

<p>The
following are suggest ions for preventing Social Engineering:</p>
<p>1.    Do Not Rely Upon Common Internal Identifiers<br>
2.      Implement a Call Back Procedure When Disclosing Protected Information<br>
3.      Implement a Security Awareness Program<br>
4.      Identify Direct Computer Support Analysts<br>
5.      Create A Security Alert System<br>
6.      Social Engineering to Test Security Policies
<a name="_Toc406789613"></a></p>

<h3>
<A NAME="Heading14">1. Do Not Rely Upon Common Internal Identifiers</A></h3>

<p>Many
companies rely on some type of identifier to authenticate employees, e.g., an
Employee Number. Unfortunately, intruders will compile a list of valid
identifiers and use these identifiers to authenticate themselves when
challenged. Intruders easily obtain identifiers from real employees as part of
the Social Engineering attack.<br>
<br>
Companies should separate personnel functions from computer support functions
by having a separate identifier for the two activities. A separate identifier
would provide additional security to both personnel and computer activities.
<a name="_Toc406789614"></a></p>

<h3>
<A NAME="Heading15">2. Implement a Call Back Procedure When Disclosing Protected
Information</A></h3>

<p>One
may have prevented many of the Social Engineering attacks by verifying the
```

```
identity of the caller by calling them back at their proper telephone number,
as listed in the company telephone directory. This procedure creates a minimal
inconvenience to legitimate activities, while providing enhanced security for
the company sensitive information. Requiring employees to call back anyone
asking for personal or proprietary information minimize compromises. Use of
Caller ID services might also be an acceptable solution.
<a name="_Toc406789615"></a></p>

<h3>
<A NAME="Heading16">3. Implement a Security Awareness Program</A></h3>

<p>Computer
professionals cannot assume that basic security practices are basic to
non-computer professionals. A good security awareness program implemented at
minimal cost can save a company from loosing millions of dollars. Without a
good security awareness program, non-computer professionals would not know that
giving out one's password to a stranger is a security risk to the company's
information.
<a name="_Toc406789616"></a></p>

<h3>
<A NAME="Heading17">4. Identify Direct Computer Support Analysts</A></h3>

<p>Every
employee of a company should have one computer analyst be the focal point for
all computer support, and should be the only person to directly contact the
employee. Each analyst should have assigned no more than 60 employees, and
instruct all employees to contact their analyst if ever contacted by someone
claiming to be from computer support.
<a name="_Toc406789617"></a></p>

<h3>
<A NAME="Heading18">5. Create A Security Alert System</A></h3>

<p>A
security alert system should be in place to allow an employee to alert other
employees of a possible intrusion. It is important to note that without this
system, intruders are able to continue the same Social Engineering attacks.
Security alert system is paramount to barring easy access for intruders.
<a name="_Toc406789618"></a></p>

<h3>
<A NAME="Heading19">6. Social Engineering to Test Security Policies</A></h3>

<p>The
only conceivable method to test security policies and their effectiveness is by
using Social Engineering. There are many security assessments to test for
physical and electronic vulnerabilities, but few assessments study the human
vulnerabilities inherent in computer users. Note that only qualified and
trusted people should perform these attacks.
<a name="_Toc396533012"></a><a name="_Toc396811723"></a><a name="_Toc397381623"></a><a
name="_Toc397382179"></a><a name="_Toc398069435"></a><a name="_Toc398069620"></a><a
name="_Toc398075274"></a><a name="_Toc401601909"></a><a name="_Toc402709927"></a><a
name="_Toc402888104"></a><a name="_Toc403222585"></a><a name="_Toc403316821"></a><a
name="_Toc403805152"></a><a name="_Toc404008210"></a><a name="_Toc404342527"></a><a
name="_Toc404342579"></a><a name="_Toc404437801"></a><a name="_Toc404438886"></a><a
name="_Toc406789619"></a></p>
```

```
<h2>
<A NAME="Heading20">E. PREVENTING PASSWORD CRACKING</A></h2>

<p>The
best ways to prevent password cracking are to create difficult-to-guess
passwords, and to protect the passwords and the file that store passwords.
Refer to Section II.1.c and Section II.1.d, for suggestions on how to choose
good passwords and how to protect passwords, respectively.
<a name="_Toc406789620"></a></p>

<h1>
<A HREF="TestTool.html" NAME="Heading21">III. TAKE TEST</A></h1>

</body></html>
```

APPENDIX C. TEST TOOL DRIVER HTML LISTING

```
<html>
<body>
<applet code=TestTool.class height=450 width=450>
</applet>
</body>
</html>
```

APPENDIX D. TEST TOOL JAVA SOURCE CODE LISTING

```
//
//     Project:   TestTool.java
//
// Description:   Test tool portion of thesis tutorial on social engineering and
//             weak passwords
//
//      Author:   Dagohoy H. Anunciado
//
//        Date:   09 November 1997
//
import java.applet.Applet;
import java.awt.*;

public class TestTool extends Applet {
    public static final int NUM_QUESTIONS = 7;
    public static final int MAX_CHECKBOXS = 5;
    public static final int MAX_TEXT_WIDTH = 50;
    public static final float PASSINGGRADE = 70.0f;

    CardLayout testToolDeck = new CardLayout();

    Button nextButton = new Button("Next");
    Button finishButton = new Button("Finish");

    Panel borderPanel[] = new Panel[NUM_QUESTIONS];
    Panel questionPanel[] = new Panel[NUM_QUESTIONS];
    Panel finalBorderPanel;
    Panel finalPanel;
    Panel questionCards = new Panel();

    CheckboxGroup cbg[] = new CheckboxGroup[NUM_QUESTIONS];
    TextArea question[] = new TextArea[NUM_QUESTIONS];
    TextArea finalTextArea;
    Checkbox answerBox[][] = new Checkbox[NUM_QUESTIONS][MAX_CHECKBOXS];
    Checkbox answerKey[] = new Checkbox[NUM_QUESTIONS];

    String resultText;
    int totalPossibleScore = 0;
    int score = 0;
    float percent = (float)0.0;


    public static void main(String args[]) {
        TestToolFrame app
          = new TestToolFrame ("TestTool Application Window");
        app.resize(450,450);
        app.show();
        app.applet.start();
    }

    public void init() {
        int i = 0, j = 0, k = 0, l = 0;
        int question_index = 0;

        GridBagLayout gl = new GridBagLayout();
```

```
GridBagConstraints gb = new GridBagConstraints();

setLayout(gl);

i = 0;
// Question 1.
j = 0;
question[i] =
    new TextArea(
        "Question 1.\n" +
        "\n" +
        "True or false.  The definition of computer security\n" +
        "is the tools, policies, procedures, and protocols used\n" +
        "to protect information stored and processed on\n" +
        "a computer system.",
        7,              // Number of question lines plus 1
        MAX_TEXT_WIDTH);

question[i].setEditable(false);
cbg[i] = new CheckboxGroup();
answerKey[i] = // Correct answer A.
answerBox[i][j] = new Checkbox("A. True", cbg[i], true); j++;
 answerBox[i][j] = new Checkbox("B. False",cbg[i], false); j++;


i++;

 // Question 2.
j = 0;
question[i] =
    new TextArea(
        "Question 2.\n" +
        "\n" +
        "What are the two most successful methods used to intrude\n" +
        "on a computer system's security?\n" +
        "  I. Social Engineering\n" +
        " II. Exploiting weak passwords\n" +
        "III. Sharing passwords\n" +
        " IV. Leaving computer logged into one's account\n" +
        "  V. Exploiting an operating system vulnerability",
        10,             // Number of question lines plus 1
        MAX_TEXT_WIDTH);

question[i].setEditable(false);
cbg[i] = new CheckboxGroup();
answerKey[i] = // Correct answer A.
 answerBox[i][j] = new Checkbox("A. I & II", cbg[i], true); j++;
 answerBox[i][j] = new Checkbox("B. II & III",cbg[i], false); j++;
 answerBox[i][j] = new Checkbox("C. IV & V",cbg[i], false); j++;
 answerBox[i][j] = new Checkbox("D. I & III",cbg[i], false); j++;
 answerBox[i][j] = new Checkbox("E. II & V",cbg[i], false); j++;
i++;

 // Question 3.
j = 0;
question[i] =
    new TextArea(
        "Question 3.\n" +
        "\n" +
        "Select the best answer from A through E.  What attributes\n" +
        "make up a good password?\n" +
        "  I. Easy to remember\n" +
        " II. Contain acronyms common to a specific field or profession\n" +
        "III. Difficult to guess\n" +
        " IV. Have digits and punctuation characters as well as letters\n" +
        "  V. Are a variation of a first or a last name",
        10,            // Number of question lines plus 1
        MAX_TEXT_WIDTH);

question[i].setEditable(false);
cbg[i] = new CheckboxGroup();
 answerBox[i][j] = new Checkbox("A. I, II, & IV", cbg[i], true); j++;
answerKey[i] = // Correct answer B.
 answerBox[i][j] = new Checkbox("B. I, III, & IV", cbg[i], false); j++;
 answerBox[i][j] = new Checkbox("C. I, IV, & V",cbg[i], false); j++;
 answerBox[i][j] = new Checkbox("D. None of the above", cbg[i], false); j++;
 answerBox[i][j] = new Checkbox("E. All of the above", cbg[i], false); j++;

i++;

 // Question 4.
j = 0;
question[i] =
    new TextArea(
        "Question 4.\n" +
        "\n" +
        "Select the best answer from A through E.  What attributes\n" +
        "make up weak password?\n" +
        "  I. Have both uppercase and lowercase letters\n" +
        " II. Are in a dictionary of some language\n" +
        "III. Contain no password at all\n" +
        " IV. Combine short words with a special character or a number\n" +
        "  V. Contain a proper noun, e.g., the name of a real or\n" +
        "     fictitious character or place",
        11,            // Number of question lines plus 1
        MAX_TEXT_WIDTH);

question[i].setEditable(false);
cbg[i] = new CheckboxGroup();
 answerBox[i][j] = new Checkbox("A. I, IV, & V", cbg[i], true); j++;
 answerBox[i][j] = new Checkbox("B. I, II, & IV", cbg[i], false); j++;
answerKey[i] = // Correct answer C.
 answerBox[i][j] = new Checkbox("C. II, III, & V",cbg[i], false); j++;
 answerBox[i][j] = new Checkbox("D. None of the above", cbg[i], false); j++;
 answerBox[i][j] = new Checkbox("E. All of the above", cbg[i], false); j++;

i++;

 // Question 5.
j = 0;
question[i] =
    new TextArea(
        "Question 5.\n" +
        "\n" +
        "True or false. Social Engineering is the process\n" +
        "of using social interactions to obtain information\n" +
        "about a company's computer system?",
        6,             // Number of question lines plus 1
        MAX_TEXT_WIDTH);

question[i].setEditable(false);
cbg[i] = new CheckboxGroup();
answerKey[i] = // Correct answer A.
 answerBox[i][j] = new Checkbox("A. True", cbg[i], true); j++;
 answerBox[i][j] = new Checkbox("B. False", cbg[i], false); j++;

i++;

 // Question 6.
j = 0;
question[i] =
    new TextArea(
        "Question 6.\n" +
        "\n" +
        "Select the best answer from A through E.\n" +
        "Social Engineering uses which of the following methods?\n" +
        "  I. Impersonating system or security administration staff\n" +
        "     for information gathering\n" +
        " II. Blackmail\n" +
        "III. Physical threat\n" +
        " IV. Rummaging through a company's garbage\n" +
        "  V. Become an employee of the company for the purpose\n" +
        "     of gathering information",
        12,            // Number of question lines plus 1
        MAX_TEXT_WIDTH);

question[i].setEditable(false);
cbg[i] = new CheckboxGroup();
 answerBox[i][j] = new Checkbox("A. I & II", cbg[i], true); j++;
 answerBox[i][j] = new Checkbox("B. II & III", cbg[i], false); j++;
 answerBox[i][j] = new Checkbox("C. I, IV, & V", cbg[i], false); j++;
 answerBox[i][j] = new Checkbox("D. None of the above", cbg[i], false); j++;
answerKey[i] = // Correct answer E.
 answerBox[i][j] = new Checkbox("E. All of the above", cbg[i], false); j++;

i++;

 // Question 7.
j = 0;
question[i] =
    new TextArea(
        "Question 7.\n" +
        "\n" +
        "Is it proper to share passwords?",
        4,             // Number of question lines plus 1
        MAX_TEXT_WIDTH);

question[i].setEditable(false);
cbg[i] = new CheckboxGroup();
 answerBox[i][j] = new Checkbox("A. Generally yes", cbg[i], true); j++;
 answerBox[i][j] = new Checkbox("B. Generally no", cbg[i], false); j++;
 answerBox[i][j] = new Checkbox("C. Yes", cbg[i], false); j++;
answerKey[i] = // Correct answer D.
 answerBox[i][j] = new Checkbox("D. No", cbg[i], false); j++;
```

```
        answerBox[i][j] = new Checkbox("E. Depends on the site's policy", cbg[i],
false); j++;

        i++;

        totalPossibleScore = i;

        // Setup GUI Layout
        questionCards.setLayout(testToolDeck);

        gb.anchor = GridBagConstraints.WEST;
        gb.gridwidth = 2;
        for (k = 0; k < i; k++) {
            borderPanel[k] = new Panel();
            borderPanel[k].setLayout(new BorderLayout());

            questionPanel[k] = new Panel();
            questionPanel[k].setLayout(gl);

            gb.gridx = 0;
            gb.gridy = 0;

            gl.setConstraints(question[k], gb);
            questionPanel[k].add(question[k]);

            gb.gridy++;

            for (l = 0; l < MAX_CHECKBOXS; l++) {
                if (answerBox[k][l] != null) {
                    gl.setConstraints(answerBox[k][l], gb);
                    questionPanel[k].add(answerBox[k][l]);
                    gb.gridy++;
                } else {
                    break;
                }
            }

            borderPanel[k].add("Center", questionPanel[k]);
            borderPanel[k].add("West",
                (k != i-1)?(new Button("Next")):
                           (new Button("Finish")));

            questionCards.add("Q" + k, borderPanel[k]);
        }

        //
        // Final Panel: Test Results
        //
        gb.anchor = GridBagConstraints.WEST;
        gb.gridwidth = 2;

        finalBorderPanel = new Panel();
        finalBorderPanel.setLayout(new BorderLayout());
        finalPanel = new Panel();
        finalPanel.setLayout(gl);

        gb.gridx = 0;
        gb.gridy = 0;
        resultText = "Test Result\n\n";

        finalTextArea =
            new TextArea(
                resultText,
                7,              // Number of question lines plus 1
                MAX_TEXT_WIDTH);

        finalTextArea.setEditable(false);

        gl.setConstraints(finalTextArea, gb);
        finalPanel.add(finalTextArea);

        questionCards.add("Results", finalPanel);

        add(questionCards);
    }

    public void start() {
        repaint();
    }

    public boolean mouseDown(Event event, int x, int y) {
// Used for debugging purposes.
//
//      testToolDeck.next(questionCards);
        return true;
    }

    public boolean action(Event e, Object o) {
        boolean result = false;
        int i = 0;

        if (e.target instanceof Button) {
            String s = (String)o;
            if ( !s.equals("Finish") ) {
                testToolDeck.next(questionCards);
            } else {
                // Calculate test results
                for (i = 0; i < NUM_QUESTIONS; i++) {
                    if (answerKey[i].getState() == true) {
                        score++;
                    }
                }

                percent = 100 * score/((float)totalPossibleScore);

                resultText
                    = "Correct Answers: " +
                      score + " of " + totalPossibleScore + "\n" +
                      "Percent Correct: " + percent + "\n\n" +
                      ((percent > PASSINGGRADE)?
                      "Congratulation you received a passing grade!\n":
                      "Please review the material and retake the test.\n");

                finalTextArea.appendText(resultText);
```

```
              testToolDeck.last(questionCards);

              score = 0;
           }
           result = true;
       }

       return result;
    }
}

class TestToolFrame  extends Frame {
   TestTool applet;

   public TestToolFrame (String frameName) {
        super(frameName);
        applet = new TestTool();
        add("Center",applet);
        applet.init();
   }

   public boolean handleEvent(Event event) {
        if(event.id == Event.WINDOW_DESTROY) {
             applet.stop();
             applet.destroy();
             System.exit(0); }
        return false;
   }
}
```

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center..2
8725 John J. Kingman Rd., Ste 0944
Ft. Belvoir, Virginia 22060-6218

2. Dudley Know Library...2
Naval Postgraduate School
411 Dyer Rd.
Monterey, California 93943-5101

3. Dr. Dan Boger, Acting Chairman, Code CS 2
Naval Postgraduate School
Monterey, California 93943-5000

4. Bert Lundy, Code CS...2
Naval Postgraduate School
Monterey, California 93943-5000

5. Ron Broersma...1
SPAWARSYSCEN San Diego
San Diego, California 92152-5000

6. Miriam Glorioso...1
SPAWARSYSCEN San Diego D42
53140 Gatchell Road
San Diego, California 92152-5000

7. Crisamar J. Anunciado 2
5903 Bataan Circle
San Diego, California 92139-1523

8. Dagohoy H. Anunciado...1
SPAWARSYSCEN San Diego D4222
53140 Gatchell Road
San Diego, California 92152-7466